
LightDB Enterprise Manager

发行版本 23.4

LightDB

2024 年 01 月 10 日

目录:

1	引言	2
1.1	编写目的	2
1.2	预期读者	2
1.3	参考文献	2
2	LightDB 安装	2
3	LightDB EM 功能介绍	2
3.1	登录页面	2
3.2	首页	3
3.3	功能模块	4
3.4	web 命令行工具	4
3.5	执行计划解析器	5
3.6	集群管控	7
3.7	集群拓扑图	24
3.8	主机监控与管理	25
3.9	数据库监控与管理	36
3.10	集中备份	69
3.11	日志检索	79
3.12	统一 sql 代理	93
3.13	通知中心	101
3.14	系统管理	104
3.15	告警待办	113
3.16	用户中心	115
4	LightDB EM 注意事项	121
4.1	Lightdb EM 集群相关	121
5	EM 参数	121
6	AGENT 参数	123
7	EM 错误码	125
8	版本发布历史	147
8.1	版本发布时间线	147
8.2	发布记录	148

1 引言

1.1 编写目的

本文档为恒生电子股份有限公司 LightDB Enterprise Manager 用户手册说明书，本文档主要阐述 LightDB Enterprise Manager 的详细功能介绍，完整的数据库功能请参考《LightDB 用户手册》，完整 LightDB Enterprise Manager 安装过程请参考安装包中的 readme.md。LightDB Enterprise Manager（即 LightDB 数据库监控管理平台，下文均简称为 LightDB EM）是一个综合性的数据库监控和管理系统，旨在满足数据库用户的需求，提供强大的图形界面，简化了对 LightDB 数据库的维护和使用。

1.2 预期读者

数据库管理员开发工程师测试工程师技术支持工程师

1.3 参考文献

《LightDB 数据库安装手册》

2 LightDB 安装

安装完整的 LightDB 数据库请参考 [LightDB 数据库安装手册](#)；

3 LightDB EM 功能介绍

3.1 登录页面

LightDB EM 默认登录地址 http://your_em_server_ip:17331/em/login.html。首次成功安装，登录用户名是 system，密码是 hs123456。首次成功安装登录进去后会提示要修改密码。密码修改后再登录就直接进入首页。



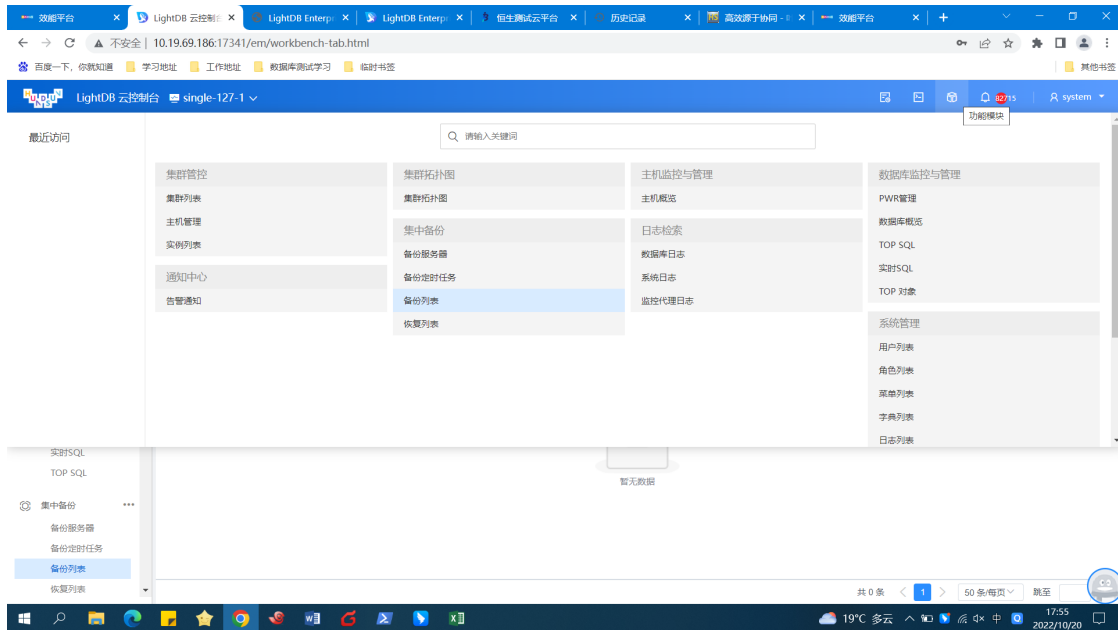
3.2 首页

登录成功后进入首页。首页为平台介绍页，包含平台功能和平台功能性描述。



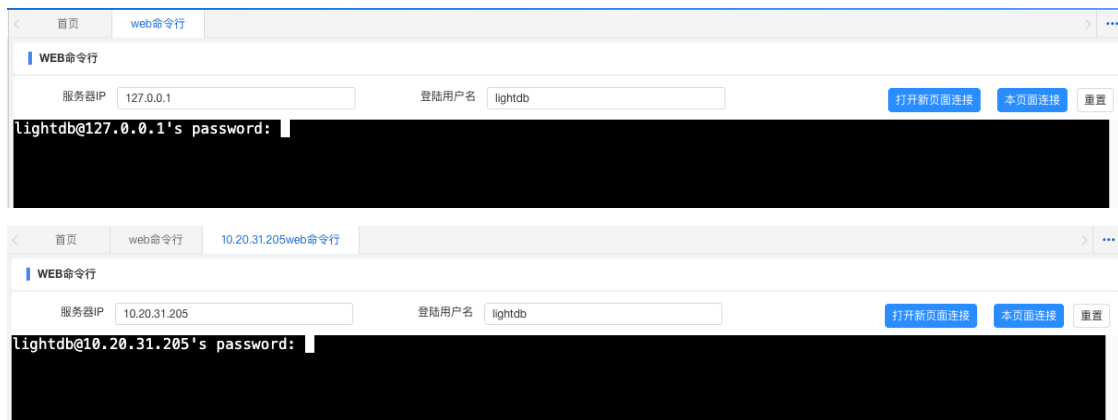
3.3 功能模块

功能模块可以展示系统所有功能清单，同时展示用户最新操作的功能，可以根据功能名称关键字查询快速查询到功能，通过点击对应的功能可以进入功能的详细页面。



3.4 web 命令行工具

Web 命令行提供一个 web 版本的命令行页面，可在此页面执行 ssh 连接到其他服务器或者其他操作。初次打开会登录到 em 所在服务器的命令行页面，需要输入对应账户的密码提供服务器 ip 和用户名输入，可通过点击打开新页面连接打开新的连接，或者本页面进行连接，点击重置在本页面将连接重置到 127.0.0.1。



3.5 执行计划解析器

执行计划解析器提供在页面顶部第一位，点击后可打开添加计划窗口



输入所需的标题和计划 json，查询语句（可选填）后，点击确定即可打开对于的执行计划解析器。

添加计划

×

* 标题

测试JSON

×

▼

* 计划json

```
Temp Read Blocks": 0,
"Temp Written Blocks": 0,
"I/O Read Time": 0.000,
"I/O Write Time": 0.000
},
"Planning Time": 11.824,
"Triggers": [
],
"Execution Time": 224.291
}
]
```

查询语句
(可选)

提示： 仅支持 JSON 格式的查询计划解析，请手动生成或从慢SQL日志获取
JSON查询计划。手动生成可使用： EXPLAIN (ANALYZE on, FORM
AT json) statement;

确认

取消

执行计划解析器页面存在三个 tab 页，分别是 html 表格解析结果，source 解析原格式，query 查询语句，html 页面中包含中间的表格数据和下方的 planing time、execution time 数据。

[首页](#)
[测试JSON执行计划解析器](#)

执行计划解析器

添加

HTML

SOURCE

序列	exclusive	inclusive	rows x	rows	loops	node
						▼WindowAgg (const=84855.41..2082520.98 rows=173000 width=144) (actual time=222.228..222.266 rows=1 loops=1) Output:row_number() OVER (?),t.fleet_id,a.service_provider_id,f.account_id,p.name,a.name,f.name,t.topic,t.trigger,(f.enabled AND a.enabled AND p.enabled),to_json(array_agg(json_build_object('id',t.id,'user_id',t.user_id,'grant_id',g.grant_id,'user',u.name,'email_address',lower(trim(COALESCE(NULLIF((u.auth0_data-->'email')::text,'')::text,(u.email_address)::text))), 'authorised',(g.grant_id IS NOT NULL),'enabled',(u.factory OR (u.enabled AND COALESCE(p_1.enabled,true) AND COALESCE((p_3.enabled AND a_2.enabled),true) AND COALESCE((p_2.enabled AND a_1.enabled AND f_1.enabled),true))), 'email_verified',COALESCE(CASE WHEN COALESCE((NOT COALESCE((u.auth0_id-->'auth0_id')::text AND ((u.auth0_data-->'user_id')::text)),false),true) THEN NULL::boolean ELSE COALESCE((u.auth0_data-->'email_verified')::text)::boolean,false) END,false),'scope',g.scope,'tenant',g.tenant,'media',t.media,'create_time_utc',t.create_time_utc,'modified_time_utc',t.modified_time_utc,'uid',t.uid))),Shared Hit Blocks:77873,Shared Read Blocks:0,Shared Dirtied Blocks:0,Shared Written Blocks:0,Local Hit Blocks:0,Local Read Blocks:0,Local Dirtied Blocks:0,Local Written Blocks:0,Temp Read Blocks:207,Temp Written Blocks:207,I/O Read Time:0,I/O Write Time:0
1	0.008	222.266	↑173000.0	1	1	
						▼Aggregate (const=84855.41..2079925.98 rows=173000 width=136) (actual time=222.221..222.258 rows=1 loops=1) Strategy:Sorted,Partial Mode:Simple,Output:t.fleet_id,a.service_provider_id,f.account_id,p.name,a.name,f.name,t.topic,t.trigger,(f.enabled AND a.enabled AND p.enabled),array_agg(json_build_object('id',t.id,'user_id',t.user_id,'grant_id',g.grant_id,'user',u.name,'email_address',lower(trim(COALESCE(NULLIF((u.auth0_data-->'email')::text,'')::text,(u.email_address)::text))), 'authorised',(g.grant_id IS NOT NULL),'enabled',(u.factory OR (u.enabled AND COALESCE(p_1.enabled,true) AND COALESCE((p_3.enabled AND a_2.enabled),true) AND COALESCE((p_2.enabled AND a_1.enabled AND f_1.enabled),true))), 'email_verified',COALESCE(CASE WHEN COALESCE((NOT COALESCE((u.auth0_id-->'auth0_id')::text AND ((u.auth0_data-->'user_id')::text)),false),true) THEN NULL::boolean ELSE COALESCE((u.auth0_data-->'email_verified')::text)::boolean,false) END,false),'scope',g.scope,'tenant',g.tenant,'media',t.media,'create_time_utc',t.create_time_utc,'modified_time_utc',t.modified_time_utc,'uid',t.uid)),Group Key:t.fleet_id,a.service_provider_id,f.account_id,p.name,a.name,f.name,t.topic,t.trigger,(f.enabled AND a.enabled AND p.enabled),Shared Hit Blocks:77873,Shared Read Blocks:0,Shared Dirtied Blocks:0,Shared Written Blocks:0,Local Hit Blocks:0,Local Read Blocks:0,Local Dirtied Blocks:0,Local Written Blocks:0,Temp Read Blocks:207,Temp Written Blocks:207,I/O Read Time:0,I/O Write Time:0
2	0.045	222.258	↑173000.0	1	1	
						▼Incremental Sort (const=84855.41..2068680.98 rows=173000 width=1004) (actual time=222.177..222.213 rows=1 loops=1) Output:t.fleet_id,a.service_provider_id,f.account_id,p.name,a.name,f.name,t.topic,t.trigger,(f.enabled AND a.enabled AND p.enabled),t.id,t.user_id,g.grant_id,u.name,u.auth0_data,u.email_address,u.factory,u.enabled,p_1.enabled,(p_3.enabled AND a_2.enabled),(p_2.enabled AND a_1.enabled AND f_1.enabled),u.auth0_id,g.scope,g.tenant,t.media,t.create_time_utc,t.modified_time_utc,t.uid,Sort Key:a.service_provider_id,f.account_id,p.name,a.name,f.name,t.topic,t.trigger,(f.enabled AND a.enabled AND p.enabled),Presorted Key:a.service_provider_id,Full-sort Groups:(object Object),Shared Hit Blocks:77873,Shared Read Blocks:0,Shared Dirtied Blocks:0,Shared Written Blocks:0,Local Hit Blocks:0,Local Read Blocks:0,Local Dirtied Blocks:0,Local Written Blocks:0,Temp Read Blocks:207,Temp Written Blocks:207,I/O Read Time:0,I/O Write Time:0
3	0.023	222.213	↑173000.0	1	1	
						▼Nested Loop (const=16529.68..1934882.3 rows=173000 width=1004) (actual time=208.24..222.19 rows=1 loops=1) Join Type:Left,Output:t.fleet_id,a.service_provider_id,f.account_id,p.name,a.name,f.name,t.topic,t.trigger,(f.enabled AND a.enabled AND p.enabled),t.id,t.user_id,g.grant_id,u.name,u.auth0_data,u.email_address,u.factory,u.enabled,p_1.enabled,(p_3.enabled AND a_2.enabled),(p_2.enabled AND a_1.enabled AND f_1.enabled),u.auth0_id,g.scope,g.tenant,t.media,t.create_time_utc,t.modified_time_utc,t.uid,Inner Unique:false,Shared Hit Blocks:77873,Shared Read Blocks:0,Shared Dirtied Blocks:0,Shared Written Blocks:0,Local Hit Blocks:0,Local Read Blocks:0,Local Dirtied Blocks:0,Local Written Blocks:0,Temp Read Blocks:207,Temp Written Blocks:207,I/O Read Time:0,I/O Write Time:0 ▼Nested Loop (const=16529.43..1931422.05 rows=173 width=950) (actual time=195.6..209.549 rows=1 loops=1)
4	0.006	222.19	↑173000.0	1	1	

Planning time : 11.824 ms
 Execution time :224.291 ms

Planning time : 11.824 ms
Execution time :224.291 ms

执行计划解析器 html 中存在 7 列数据，分别是序列号，exclusive 独占时间、inclusive 所有时间，rowsx 行数 x，rows 行数，loops 循环数，node 节点信息。前四个节点有四种背景颜色，依次为红色，棕色，黄色，白色，分别对应严重到不严重的等级，带有↓的值表示计划被低估了给定的次数，带有↑的值表示它被高估了。其中 exclusive，inclusive 的比例取决当前数值占总时间的比例，依次为 90%/50%/10%，rowsx 依据当前值的大小依次为 1000/100/10，rows 取决的移除数的比例，计算规则为 rows/ (rows+remove)，比例依次为 90%/50%/10%。同时提供自定义展示值的功能，可自定义解析出的展示结果

3.6 集群管控

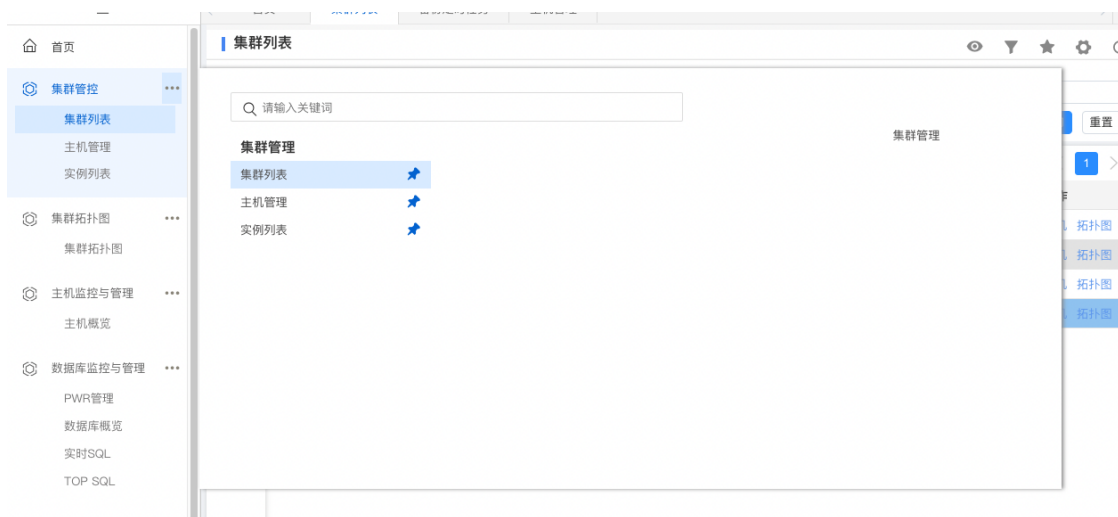
集群管控用于统一管理主机、代理、实例

集群列表

通过添加集群来管控集群下的主机和数据库实例。点击新增主机可跳转至主机管理。点击拓扑图可跳转至集群拓扑图查看集群下主机角色关联关系。

查询集群列表

集群列表展示用户添加的集群信息，可以根据部署模式、数据库类型、集群名称、首选集群进行查询，同时提供增删改操作功能。



添加集群

用户可以添加要监控的集群，添加成功后可通过添加主机、实例进行关联。

注意：EM 添加集群初始化实例仅用于开发和测试人员使用，生产环境请使用 LightDB 发布的安装包或客户端工具进行操作。

The screenshot shows the '新增集群' (New Cluster) configuration window. It includes the following fields and controls:

- 数据库类型** (Database Type): 11.LightDB
- 集群名称** (Cluster Name): single
- 数据库版本** (Database Version): 待输入 (Pending Input)
- 部署模式** (Deployment Mode): 0.Single
- 首选集群** (Preferred Cluster): 0.否 (No)
- 集群LVS** (Cluster LVS): 待输入 (Pending Input)
- 集群类型** (Cluster Type): ☒ (Enabled)
- 开发者模式** (Developer Mode): ☐ (Disabled)
- 兼容模式** (Compatibility Mode): LightDB
- 字符集编码格式** (Character Set Encoding Format): GBK
- 主机IP** (Host IP): 10.20.31.205
- LTHOME**: 请选择 (Please Select)
- LTDATA**: /usr/local/lightdb/lt_data
- 新增一行** (Add Row) and **删除** (Delete) buttons for the table below.
- Table Headers**: 主机IP, LTHOME, LTDATA
- Table Content**: 暂无数据 (No Data)
- 启用归档** (Enable Archiving): ☒ (Enabled)
- 启用增量检查点** (Enable Incremental Checkpoint): ☒ (Enabled)

At the bottom right, there are **取消** (Cancel) and **确定** (Confirm) buttons.

数据库类型：要监控的集群数据库类型包含 LightDB、PostgreSQL 等。部署模式：Single 单机模式、HA 高可用模式、Distribute 分布式模式。集群名称：定义部署的集群名称，不可重复。首选集群：平台默认展示的监控集群。数据库版本：可不填，由 agent 后台监控并填充。集群 LVS：分布式预留字段，可不填写。集群类型：打开可在线安装数据库实例。开发者模式：可选是否为开发者模式。字符集编码格式：可选 UTF8,GBK,GB18030。兼容模式：可选 LightDB、Oracle、MySQL。启用归档：可选是否启用归档模式。启用增量检查点：可选是否启用增量检查点。

注意：开发者模式开启之后，启用增量检查点开关不可见，默认开启。实例化 LightDB-23.1 版本以前，不显示启用归档开关。实例化 LightDB-23.4 版本以前，不显示启用增量检查点开关。其中具体参数和含义见 [LightDB 数据库安装手册](#)。

修改集群

修改集群不可修改数据库类型、部署模式。其余均可修改。

205	Single	LightDB	0	否	新增
dev					新增
高可用					新增
205高可用					新增

修改集群

数据库类型

1:LightDB

* 集群名称

205

数据库版本

13.3-22.2

部署模式

0:Single

* 首选集群

0:否

集群LVS

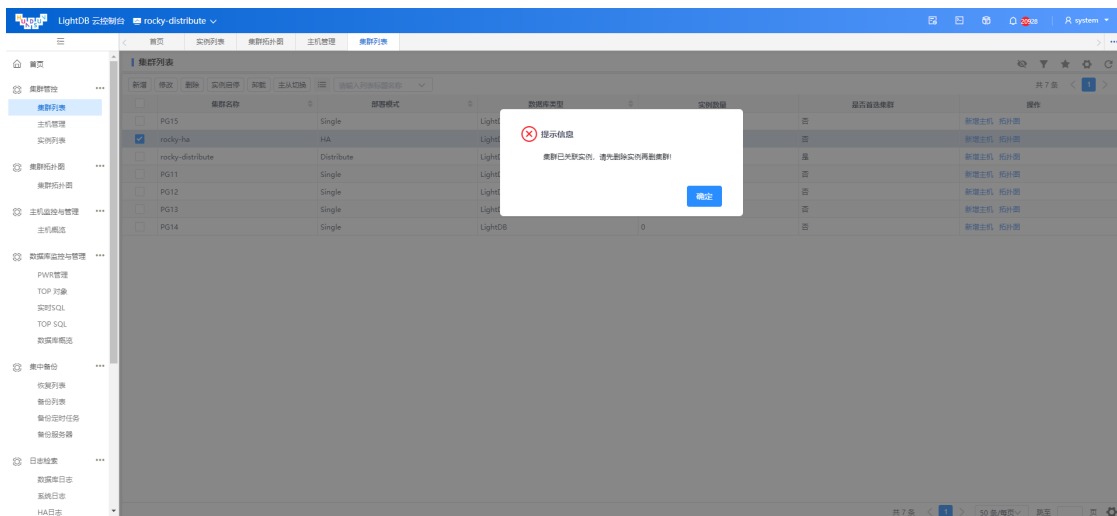
请输入

取消

保存

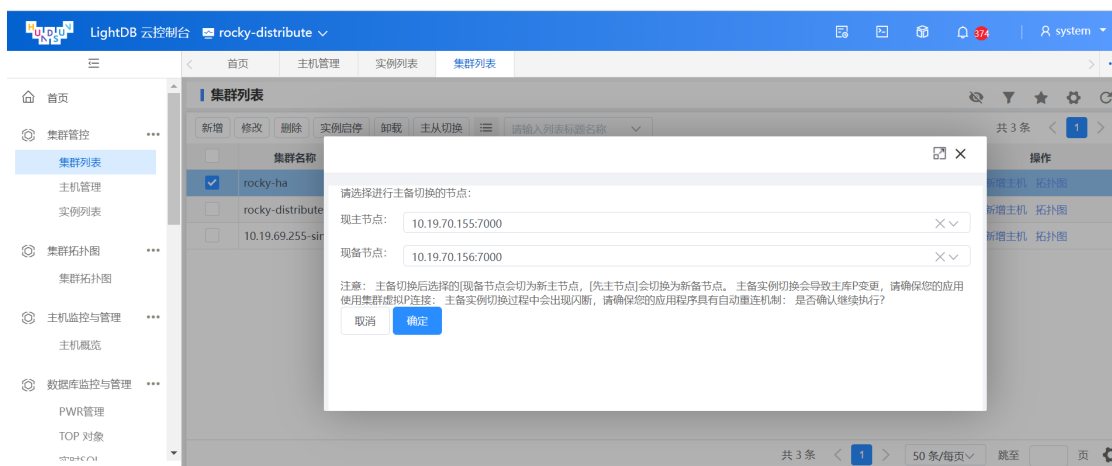
删除集群

不想要监控集群可进行删除。删除前必须先删除所监控的数据库实例。



主从切换

高可用模式下，主节点能够切换成备节点，备节点切换成主节点。



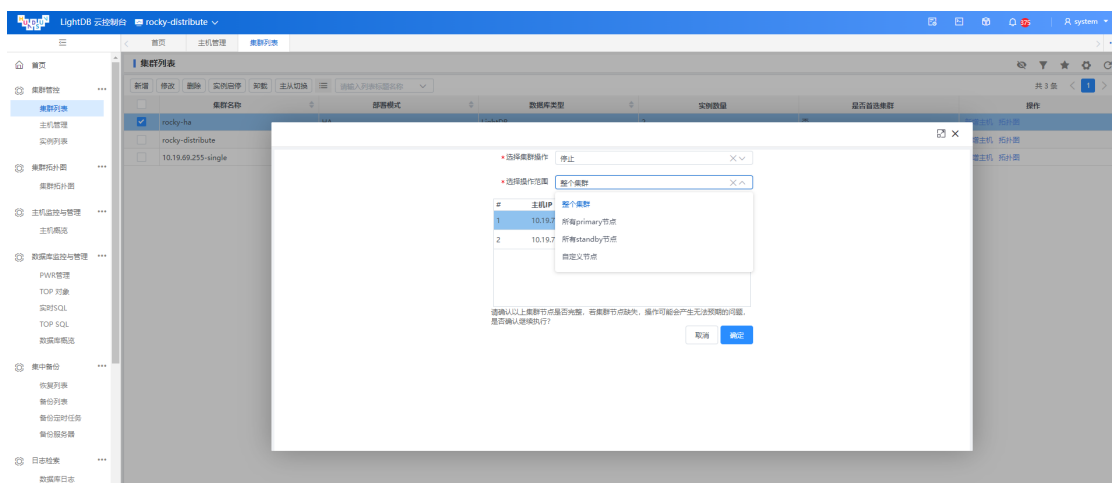
现主节点：现在高可用模式下的主节点 ip。现备节点：现在高可用模式下的备节点 ip。

实例启停

该功能需要配置 SSH 免密互通

```
# 免密认证，所有服务器都要执行
ssh-keygen -t rsa -P "" -f ~/.ssh/id_rsa
# 证书同步，所有服务器都要执行，包括自身，例如
# 192.168.10.128服务器上执行
ssh-copy-id lightdb@192.168.10.128
ssh-copy-id lightdb@192.168.10.110
# 192.168.10.110服务器上执行
ssh-copy-id lightdb@192.168.10.110
ssh-copy-id lightdb@192.168.10.128
```

实例启停操作可对整个集群中的节点进行启动，停止，重启操作。



集群卸载

The screenshot shows the 'Cluster List' dialog box in the LightDB console. The dialog box has a title bar with a close button. Below the title bar, there is a dropdown menu for 'Add Mode' (加群方式) set to 'Select' (选择). Below this, there is a table with the following columns: #, Host IP (主机IP), Cluster Name (集群名称), and Status (状态). The table contains three rows of data:

#	Host IP (主机IP)	Cluster Name (集群名称)	Status (状态)
1	10.20.0	仅和主实例	
2	10.20.43.157	5435 HA Primary	运行
3	10.20.43.158	5435 HA Witness	未和

At the bottom right of the dialog box, there are two buttons: 'Cancel' (取消) and 'Confirm' (确定).

主机管理

通过添加主机部署监控实例代理。点击代理状态可跳转至监控代理日志。点击概览可以跳转到主机概览。

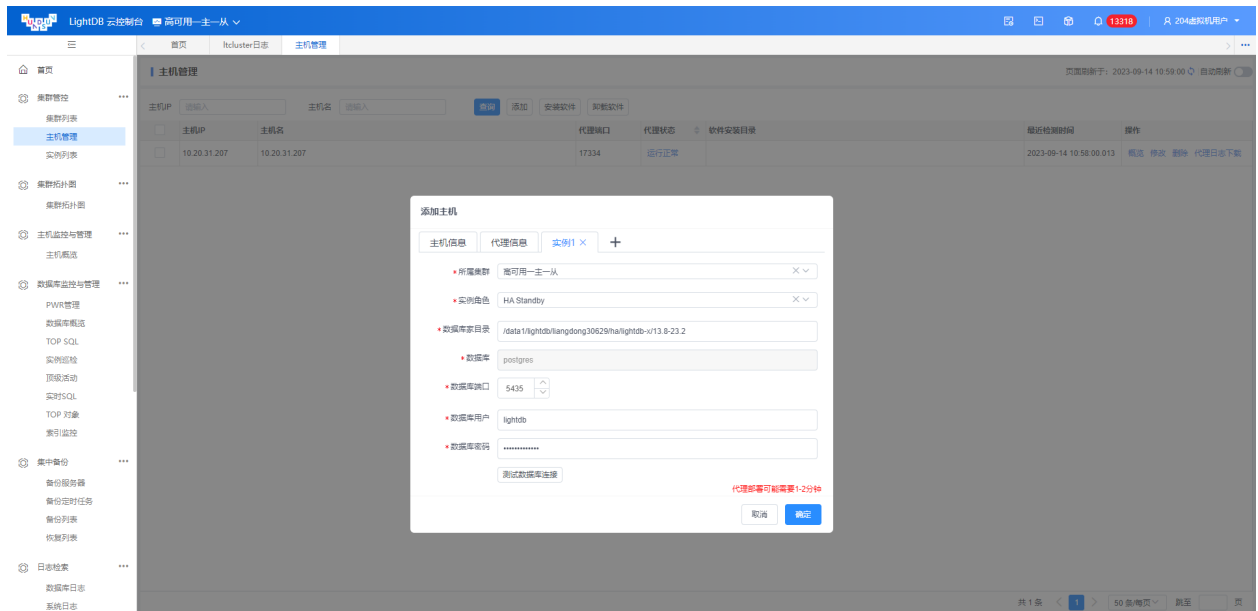
添加主机

```
/var/log/messages
{
create 0644
postrotate
/bin/kill -HUP `cat /var/run/syslogd.pid 2> /dev/null` 2> /dev/null || true
endscript
}
```

```
logrotate --force /etc/logrotate.d/syslog
```

用户可以添加要监控的主机，添加成功后系统自动把代理安装到配置的目录中

重要：当前版本，添加主机监控的同时支持添加主机上多实例监控(最多支持 4 个实例)



主机IP：主机IP地址
主机名：方便识别主机
SSH端口：主机的ssh端口
主机用户名：登录主机的用户名
主机密码：登录主机用户名对应的密码

添加主机

主机信息	代理信息	实例1 ×	+
* 主机IP	10.20.31.204		
* 主机名	10.20.31.204		
* 主机用户名	lightdb		
* 主机密码			
* SSH端口	22 ^ v		
✓ 测试SSH连接			

代理部署可能需要1-2分钟

取消

确定

代理端口：要监控主机中未使用的端口，默认17334。建议1733X未占用相关端口。
代理安装路径：代理安装的路径，需要主机用户名有RWX权限的空目录。
agent包上传：上传打包好的agent的zip包，根据服务器的版本选择对应的zip包。

添加主机

主机信息

代理信息

实例1 ×

+

* 代理端口

17334

* 代理路径

/data1/lightdb/liangdong30629/agent/host

* agent包上传


点击或将文件拖拽到这里上传

LightDB1.0-agent-V202303-00-000-el7-x86_64.zip

×

代理部署可能需要1-2分钟

取消

确定

允许在添加主机监控的同时监控实例信息

所属集群：所要监控的数据库实例对应的集群。
所属主机：所要监控的数据库实例安装在哪个主机上。
实例角色：所要监控的数据库实例对应的角色。
数据库家目录：数据库实例的家目录。默认 /usr/local/lightdb/lightdb-x/13.8-23.1
数据库：默认postgres，不可修改。
数据库端口：数据库实例对应端口。默认5434
数据库用户：登录数据库的用户名
数据库密码：登录数据库的密码

添加主机

主机信息	代理信息	实例1 ×	+
------	------	-------	---

* 所属集群

高可用一主一从

× v

* 实例角色

HA Standby

× v

* 数据库家目录

/data1/lightdb/liangdong30629/ha/lightdb-x/13.8-23.2

* 数据库

postgres

* 数据库端口

5435

^ v

* 数据库用户

lightdb

* 数据库密码

.....

测试数据库连接

代理部署可能需要1-2分钟

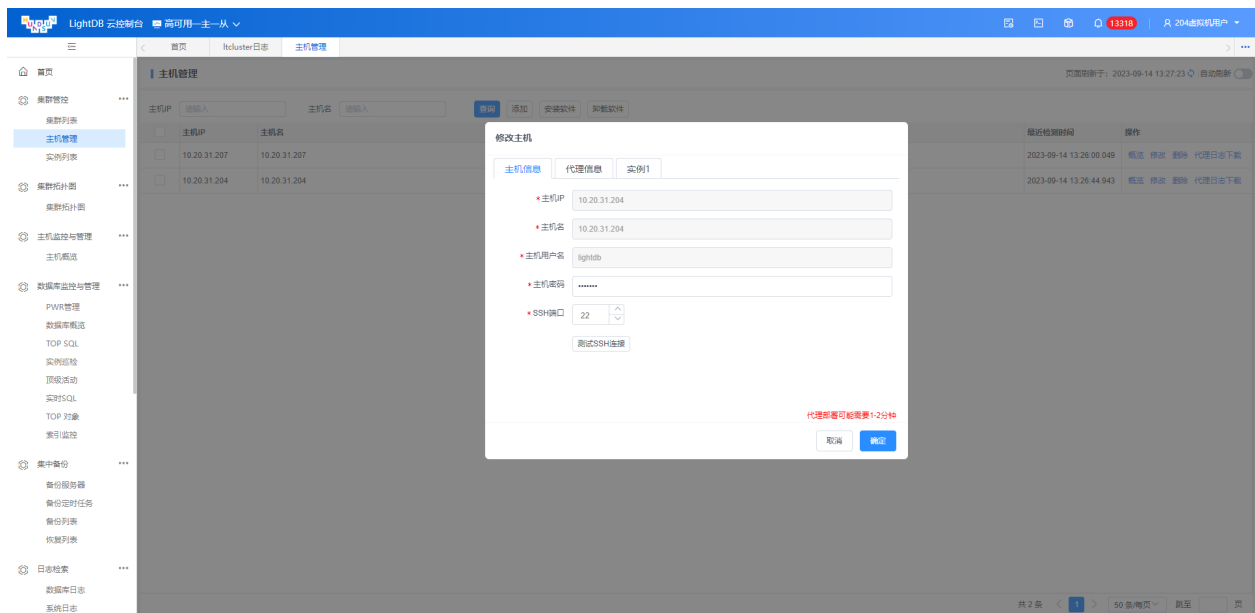
取消 确定

添加主机后代理部署可能需要 1~2 分钟，代理状态为运行正常时部署成功。

修改主机

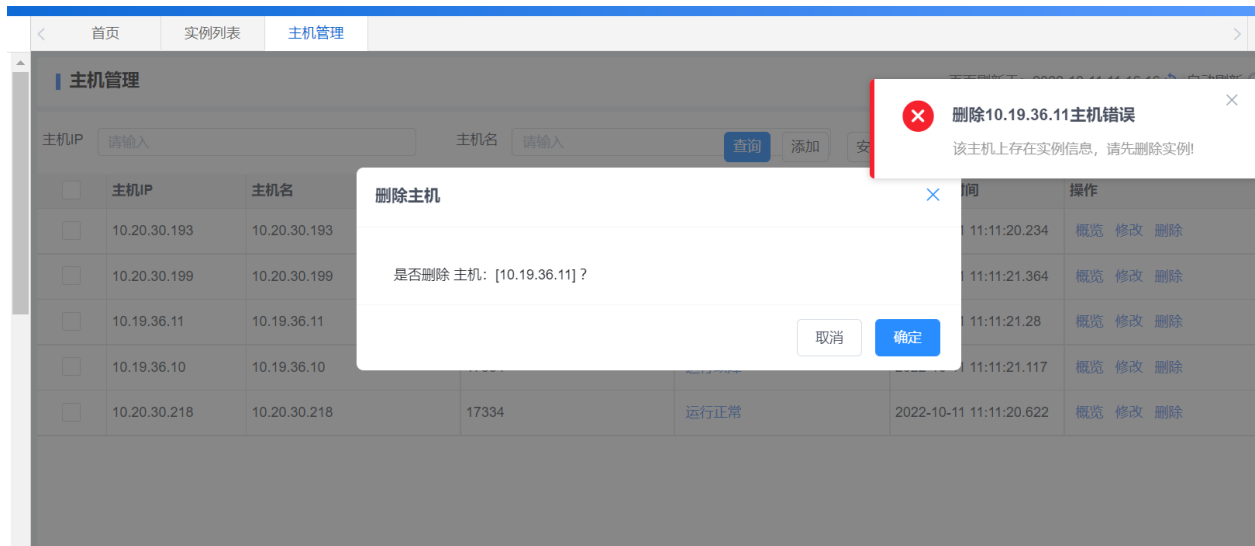
注意： 修改主机，只能修改主机密码、端口。

修改主机上多实例，目前只支持修改数据库用户密码。



删除主机

如果不需要监控可以删除主机。删除前必须先删除部署在主机上的数据库实例监控。



LightDB 云控制台 distribute

实例列表

新增 删除 修改 请输入列表标题名称

共 5 条

	所属集群	集群类型	实例角色	主机IP	DB端口	DB用户名	代理状态	代理端口	实例状态	Itcluster状态	Itcluster
☐	distribute	托管	Distribute Worker Primary	10.20.30.218	6789	lightdb	运行正常	17339	in production		
☐	distribute	托管	Distribute Worker Primary	10.19.36.11	6789	lightdb	运行正常	17339	in production		
☐	ha	托管	HA Standby	10.20.30.193	5435	lightdb	运行故障	17339	unknown	unkno...	unkno...
☐	distribute	托管	Distribute Worker Primary	10.19.36.10	6789	lightdb	运行故障	17339	unknown		
☐	distribute	托管	Distribute Worker Primary	10.20.30.199	6789	lightdb	运行正常	17340	in production		

共 5 条 1 50 条/每页 跳至 页

主机代理日志下载

在主机管理列表中，点击代理日志下载可进行下载代理日志操作。

LightDB 云控制台 HA

实例列表

新增 删除 修改 请输入列表标题名称

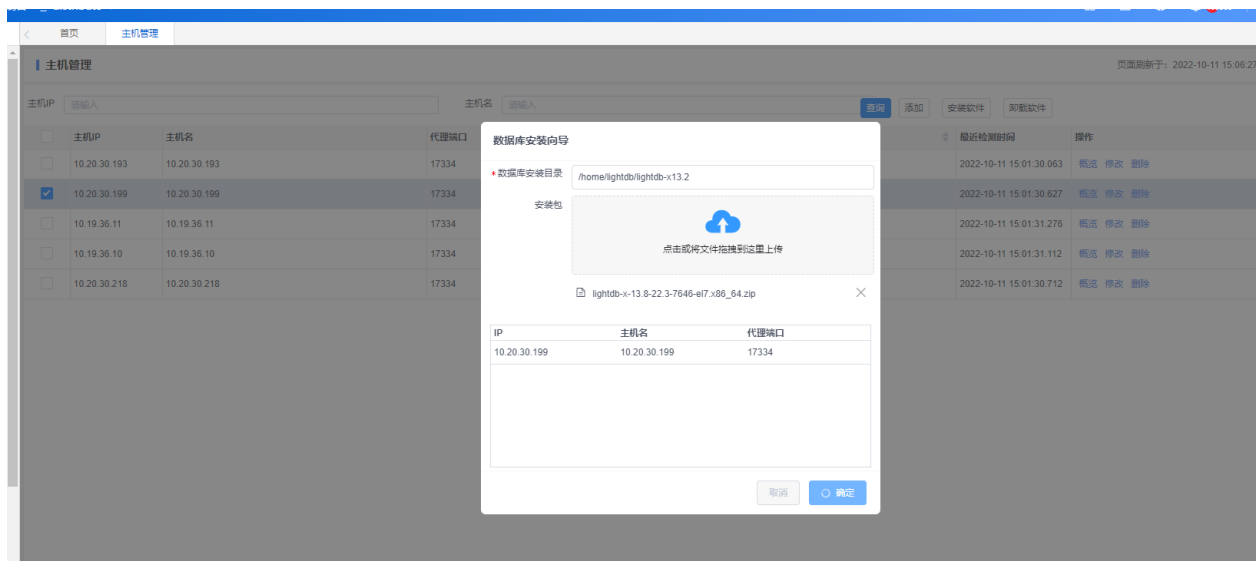
共 3 条

	集群类型	所属集群	实例角色	主机IP	DB端口	DB用户名	代理端口	代理状态	实例状态	Itcluster状态	Itcluster paused	keepalived状态	操作
☐	托管	217single	Single Only	10.20.30.217	10001	lightdb	17338	运行正常	in production	running	not paused	active	代理日志下载
☐	托管	HA	HA Primary	10.20.30.193	5438	lightdb	17338	运行正常	in production	running	not paused	active	代理日志下载
☐	托管	HA	HA Standby	10.20.30.199	5438	lightdb	17338	运行正常	in archive recovery	running	not paused	active	代理日志下载

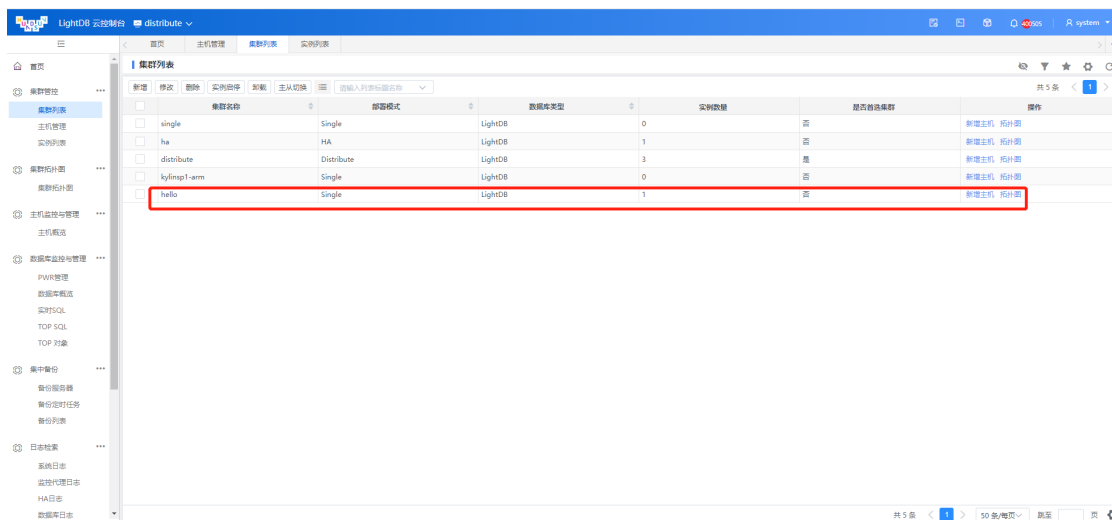
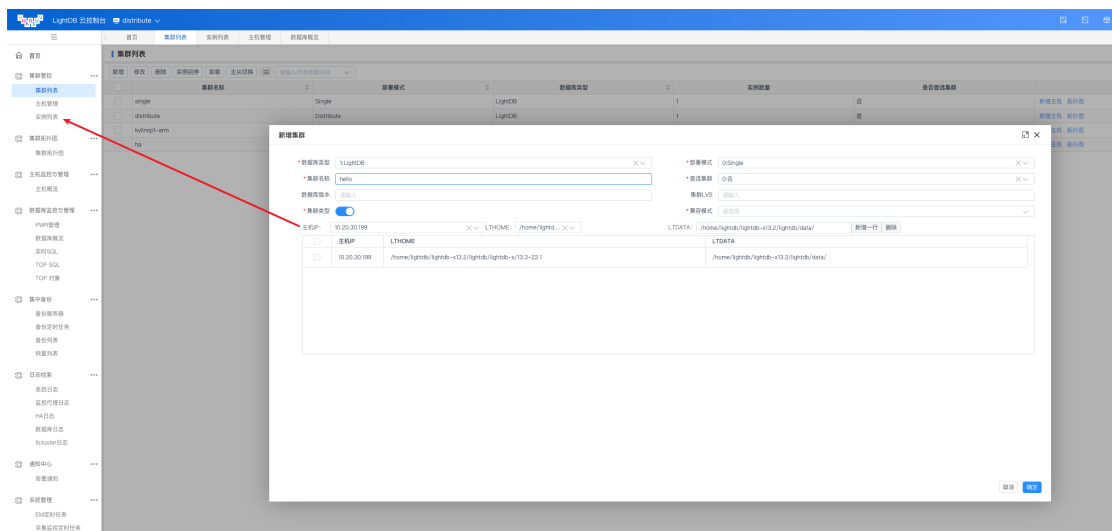
共 3 条 1 50 条/每页 跳至 页

安装软件

安装软件是指安装数据库，主机管理填写数据库安装目录，需要主机用户名有 RWX 权限的空目录，上传操作系统版本相匹配的 lightdb-x 安装包。

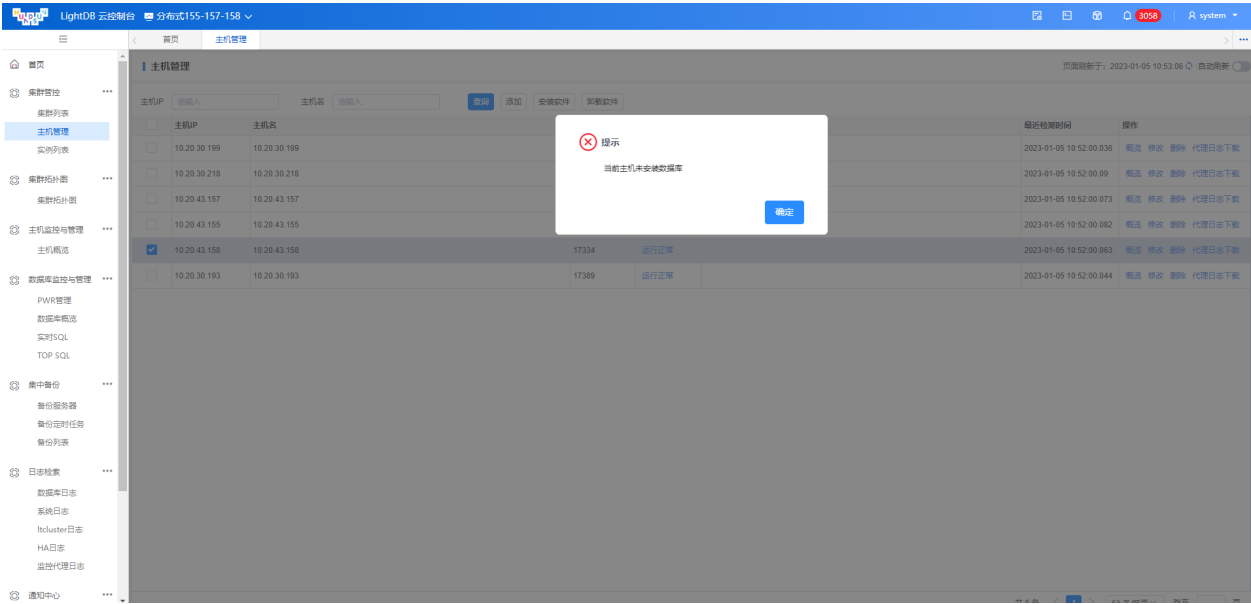


集群列表点击新增集群可以基于已安装数据库软件的主机，在线安装数据库实例。



卸载软件

卸载软件是指卸载通过 em 安装的数据库



实例列表

- 备注：**1、实例列表展示用户添加集群下主机对应部署的数据库实例信息。
- 2、包含所属集群、实例角色、主机 IP、数据库端口、数据库用户名、数据库实例状态。
- 3、可以根据所属集群、实例角色、主机 IP 和实例状态进行查询，同时提供点击概览跳转至数据库概览界面功能。

添加实例

- 重要：**1、支持添加多实例。
- 2、支持多实例和主机共用一个主机代理。
- 3、也支持新增一个只监控实例的代理。

* 是否与主机共用代理

是

否

* 主机

10.20.31.204

✕

取消

确定

添加实例, 共用一个主机代理

实例1

+

* 所属集群

204单机

✕

* 实例角色

Single Only

✕

* 数据库家目录

/data1/lightdb/liangdong30629/single2/lightdb-x/13.8-23.2

* 数据库

postgres

* 数据库端口

5433

^

v

* 数据库用户

lightdb

* 数据库密码

.....

✓ 测试数据库连接

代理部署可能需要1-2分钟

取消

确定

* 是否与主机共用代理

是

否

主机

10.20.31.204

✕

取消

确定

添加实例,实例和主机不共用一个代理

代理信息

实例1

+

* 所属集群

204单机

✕

* 实例角色

Single Only

✕

* 数据库家目录

/usr/local/lightdb/lightdb-x/13.8-23.1

* 数据库

postgres

* 数据库端口

5433

^

v

* 数据库用户

lightdb

* 数据库密码

.....

✓ 测试数据库连接

代理部署可能需要1-2分钟

取消

确定

修改实例

注意: 修改实例, 目前只支持修改单实例的数据库密码

代理信息

实例1

* 所属集群

204单机

* 所属主机

10.20.31.204

* 实例角色

Single Only

* 数据库家目录

/data1/lightdb/liangdong30629/single2/lightdb-x/13.8-23.2

* 数据库

postgres

* 数据库端口

5433

* 数据库用户

lightdb

* 数据库密码

.....

✓ 测试数据库连接

代理部署可能需要1-2分钟

取消

确定

删除实例

注意：1、删除实例前，若存在实例的备份定时任务，先删除实例的备份定时任务。2、删除实例前，若当前被监控实例是通过 em 在线安装的，请先卸载对应实例，卸载数据库软件之后，再执行删除实例操作。

LightDB 云控制台

高可用—主—从

13355

共 2048 租户用户

实例列表

删除

修改

添加新实例

实例类型	所属集群	实例角色	主机IP	数据库端口	数据库用户	代理类型	代理状态	代理端口	实例状态	Itcluster	Itcluster paused	keepalived状态	操作
☒ 托管	204单机	Single Only	10.20.31.204					17334	in production				删除
☐ 托管	单机	Single Only	10.20.31.207					17334	in production				删除
☐ 托管	高可用—主—从	HA Primary	10.20.31.207					17334	in production	running	not paused	active	删除
☐ 托管	高可用—主—从	HA Standby	10.20.31.204					17334	in archive recovery	running	not paused	active	删除

提示信息

您确定要删除选中的数据吗?

取消

确定

共 4 条

1/1

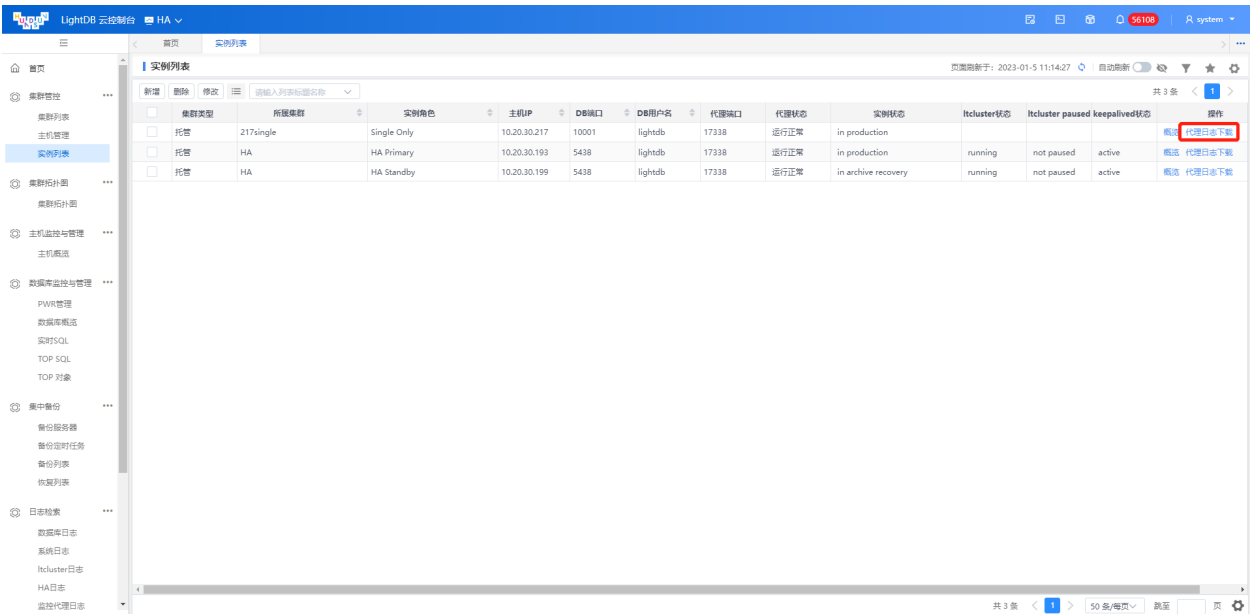
10 条/页

刷新

重置

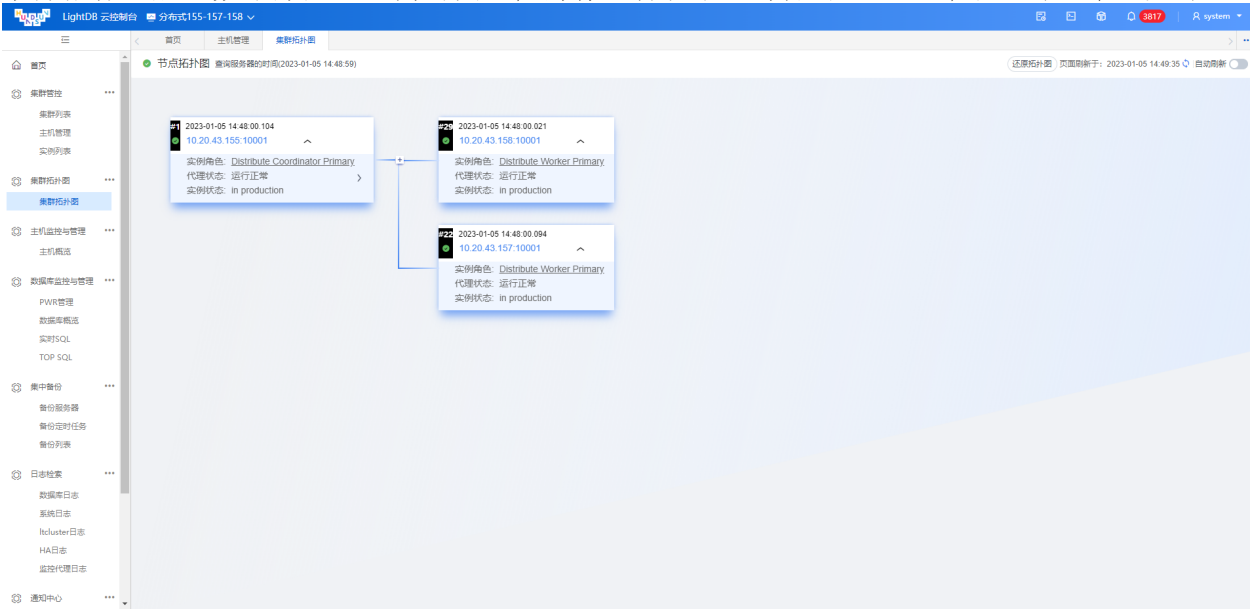
实例代理日志下载

在实例列表中，点击代理日志下载可进行下载代理日志操作。



3.7 集群拓扑图

集群拓扑图展示数据库实例之间的关系，内容包含以下三部分: 实例角色，实例状态，代理状态。



3.8 主机监控与管理

主机监控与管理展示了数据库服务器的主机概览信息。

主机概览

主机概览包括数据时间范围统一选择、自动刷新、主机信息、主机配置、CPU 使用率、网络流量分析、内存使用率、交换空间使用量、文件系统、磁盘活动和性能测试。

时间范围统一选择

时间范围统一选择提供的选择范围有 0.5 小时、1 小时、6 小时、12 小时、24 小时和自定义范围。通过选择不同的时间范围，页面下面的图表就可以展示不同时间范围的数据，除了自定义时间范围外，其他的时间范围都是以服务器当前时间作为结束时间向前推导开始时间来查询数据。



自定义范围是用户根据实际需要可以选择图表数据范围的起止时间。确定后，页面上展示选择的时间范围，同时图表数据按照选定的时间范围进行加载。首先选择日期范围

(2022-01-29 16:29:03)

时间范围 0.5小时 ^

▼

主机配置

CPU: HUAWEI,Kunp

内存: 765G

Swap: 16G

启动 | 持续 3 月 20 天 6 小时

erver V10 内核参数 >

er

<< < 2022年 1月 2022年 2月 > >>

日	一	二	三	四	五	六	日	一	二	三	四	五	六
26	27	28	29	30	31	1	30	31	1	2	3	4	5
2	3	4	5	6	7	8	6	7	8	9	10	11	12
9	10	11	12	13	14	15	13	14	15	16	17	18	19
16	17	18	19	20	21	22	20	21	22	23	24	25	26
23	24	25	26	27	28	29	27	28	1	2	3	4	5
30	31	1	2	3	4	5	6	7	8	9	10	11	12

选择时间

清空 确定

0.5小时

1小时

6小时

12小时

24小时

自定义

s0f0 ▾ 查看

日期范围选择后，可以选择时间范围

(2022-01-29 16:29:03)

时间范围 0.5小时 ^

▼

主机配置

CPU: HUAWEI,Kunp

内存: 765G

Swap: 16G

启动 | 持续 3 月 20 天 6 小时

erver V10 内核参数 >

er

0.5小时

1小时

6小时

12小时

24小时

自定义

s0f0 ▼ 查看

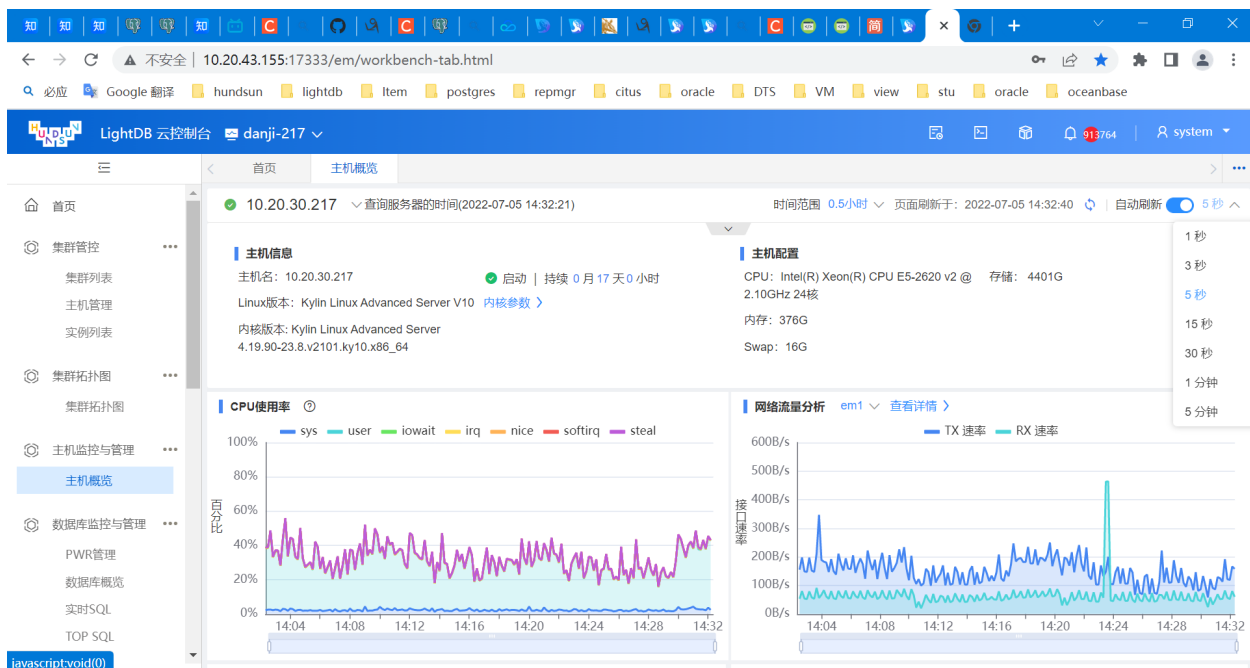
开始时间			结束时间		
00	00	00	00	00	00
01	01	01	01	01	01
02	02	02	02	02	02
03	03	03	03	03	03
04	04	04	04	04	04
05	05	05	05	05	05
06	06	06	06	06	06
07	07	07	07	07	07
08	08	08	08	08	08

选择日期

清空 确定

自动刷新

自动刷新默认是开启的且刷新频率为 5 秒，用户可以根据需要开启。自动刷新频率选择的范围有 1 秒、3 秒、5 秒、15 秒、30 秒、1 分钟和 5 分钟。开启自动刷新，选择频率后，页面会根据选择的时间频率加载图表的数据。



主机信息和主机配置

主机概览展示监控的数据库主机信息，包括主机名、Linux 版本、内核版本、启动时间和内核参数入口。

主机信息

主机名: 10.19.69.193

启动 | 持续 2 月 28 天 19 小时

Linux版本: CentOS Linux 7.6.1810

[内核参数 >](#)

内核版本: CentOS Linux 4.14.0-115.el7a.0.1.aarch64

点击内核参数入口进入内核参数列表

内核参数

参数名	参数值	建议值
aio_max_nr	65536	1048576
dirty_background_ratio	10	5
dirty_expire_centisecs	3000	500
dirty_ratio	20	40
dirty_writeback_centisecs	500	250
file_max	1607571	131072
overcommit_memory	0	2
overcommit_ratio	50	50
sem	32000 1024000000 500 32000	500 2048000 200 4096
shmall	18446744073692774399	4230005915648
shmmax	18446744073692774399	17326104230494207
shmmni	4096	4096

内核参数项介绍.

参数	说明
shmmni	整个系统的共享内存段的最大数目（个）
shmmax	内核所允许的最大共享内存段的大小（bytes）
shmall	在任何给定时刻，系统上可以使用的共享内存的总量（bytes）
sem	用于控制内核信号量，信号量是 System V IPC 用于进程间通讯的方法
swappiness	控制换出运行时内存的相对权重。swappiness 参数值可设置范围在 0 到 100 之间。低参数值会让内核尽量少用交换，更高参数值会使内核更多的去使用交换空间。默认值为 60。
overcommit_memory	内核针对内存分配的策略，其值可以是 0、1、2。0 表示内核将检查是否有足够的可用内存供应用进程使用；如果有足够的可用内存，内存申请允许；否则，内存申请失败，并把错误返回给应用进程。1 表示内核允许分配所有的物理内存，而不管当前的内存状态如何。2 表示内核允许分配超过所有物理内存和交换空间总和的内存（参照 overcommit_ratio）
overcommit_ratio	如果 overcommit_memory=2，可以过载内存的百分比，通过以下公式来计算系统整体可用内存。 系统可分配内存=交换空间+物理内存*overcommit_ratio/100
dirty_background_ratio	表示脏数据到达系统整体内存的百分比，此时触发 pdflush 进程把脏数据写回磁盘。
dirty_ratio	表示如果进程产生的脏数据到达系统整体内存的百分比，此时进程自行把脏数据写回磁盘。
dirty_expire_centisecs	表示如果脏数据在内存中驻留时间超过该值，pdflush 进程在下次将把这些数据写回磁盘。
dirty_writeback_centisecs	表示 pdflush 进程周期性间隔多久把脏数据写回磁盘。
aio-max-nr	该参数决定了系统中所允许的文件句柄最大数目。
file-max	指定了可以分配的文件句柄的最大数目。如果用户得到的错

	误消息声明由于打开文件数已经达到了最大值，从而他们不能打开更多文件，则可能需要增加该值。
--	---

27 / 89

somaxconn	定义了系统中每一个端口最大的监听队列的长度,这是个全局的参数,默认值为 1024。
tcp_max_syn_backlog	对于那些依然还未获得客户端确认的连接请求，需要保存在队列中最大数目。
tcp_tw_reuse	是否允许重新应用处于 TIME-WAIT 状态的 socket 用于新的 TCP 连接。缺省设置为 0。
tcp_syn_retries	示本机向外发起 TCP SYN 连接超时重传的次数，不应该高于 255；该值仅仅针对外出的连接,对于进来的连接由 tcp_retries1 控制。
tcp_retries2	表示放弃在已经建立通讯状态下的一个 TCP 数据包前进行重传的次数。缺省设置为 15。
tcp_slow_start_after_idle	关闭 tcp 的连接传输的慢启动，即先休止一段时间，再初始化拥塞窗口。

主机概览展示监控的数据库主机配置，包括 CPU、存储、内存和 Swap 交换区。

主机配置

CPU: HUAWEI,Kunpeng 920 96核

存储: 47896G

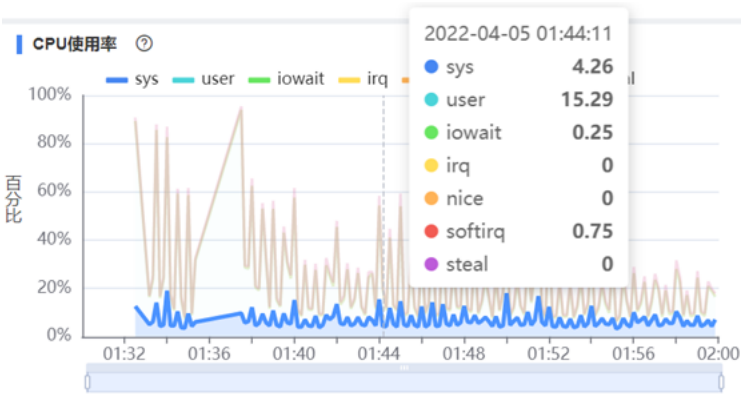
内存: 765G

Swap: 16G

CPU 使用率

目前系统每 10 秒采集一次 CPU 使用时间信息，主要采集包括 user、nice、system、idle、iowait、irq、softirq 和 steal。采集数据项介绍：

数据项	说明
user	代表用户态 CPU 时间比率。通常缩写为 us
nice	代表低优先级用户态 CPU 时间比率，也就是进程的 nice 值被调整为 1-19 之间是的 CPU 时间。通常缩写为 ni
system	代表内核态 CPU 时间比率。通常缩写为 sys
idle	代表空闲时间，它不包括 I/O 等待时间（iowait）。通常缩写为 id
iowait	代表等待 I/O 的 CPU 时间比率。通常缩写为 wa
irq	代表处理硬中断的 CPU 时间比率。通常缩写为 hi
softirq	代表处理软中断的 CPU 时间比率。通常缩写为 si
steal	代表当系统运行在虚拟机中的时候，被其他虚拟机占用的 CPU 时间比率。通常缩写为 st

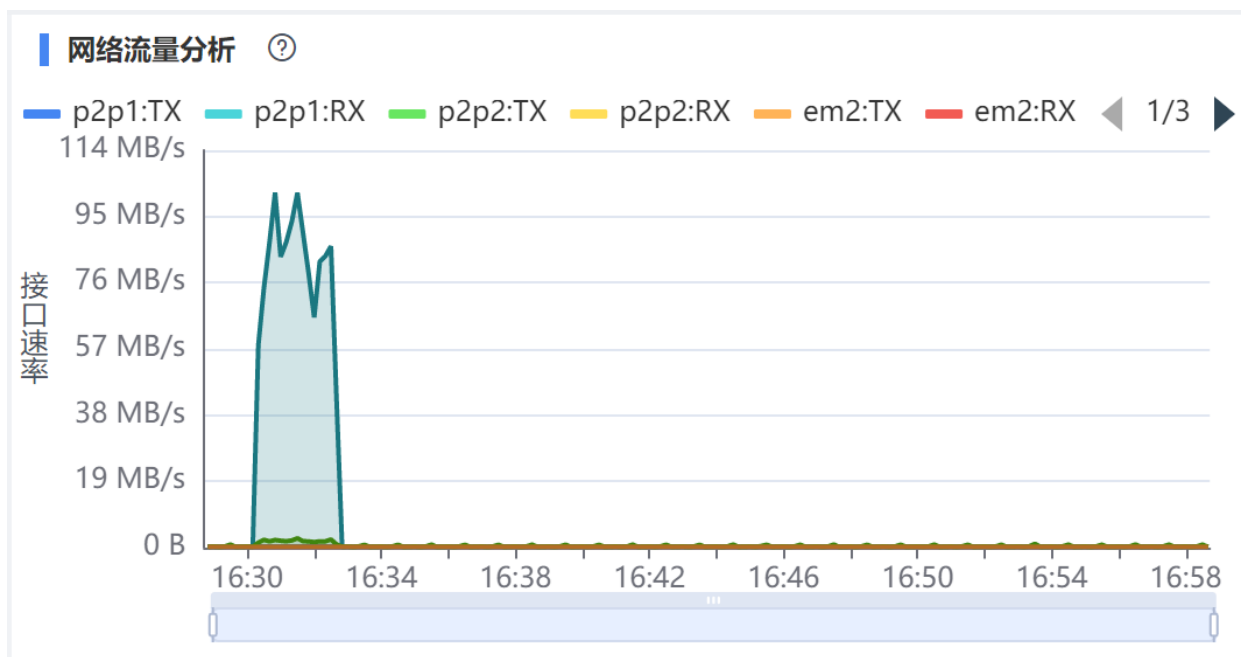


CPU 使用率计算公式，以 user 项为例： $\text{user 使用率} = \frac{\text{user}}{(\text{system} + \text{user} + \text{iowait} + \text{irq} + \text{nice} + \text{softirq} + \text{steal})}$

网络流量分析

网络流量每 10 秒采集一次，采集每个网卡的发送和接受流量速率。采集数据项介绍：

数据项	说明
TX 速率	从网卡发出的流量速率
RX 速率	网卡接收流量速率



同时提供针对网卡进程的实时流量监控，页面端提供自动刷新功能：

10.19.36.10

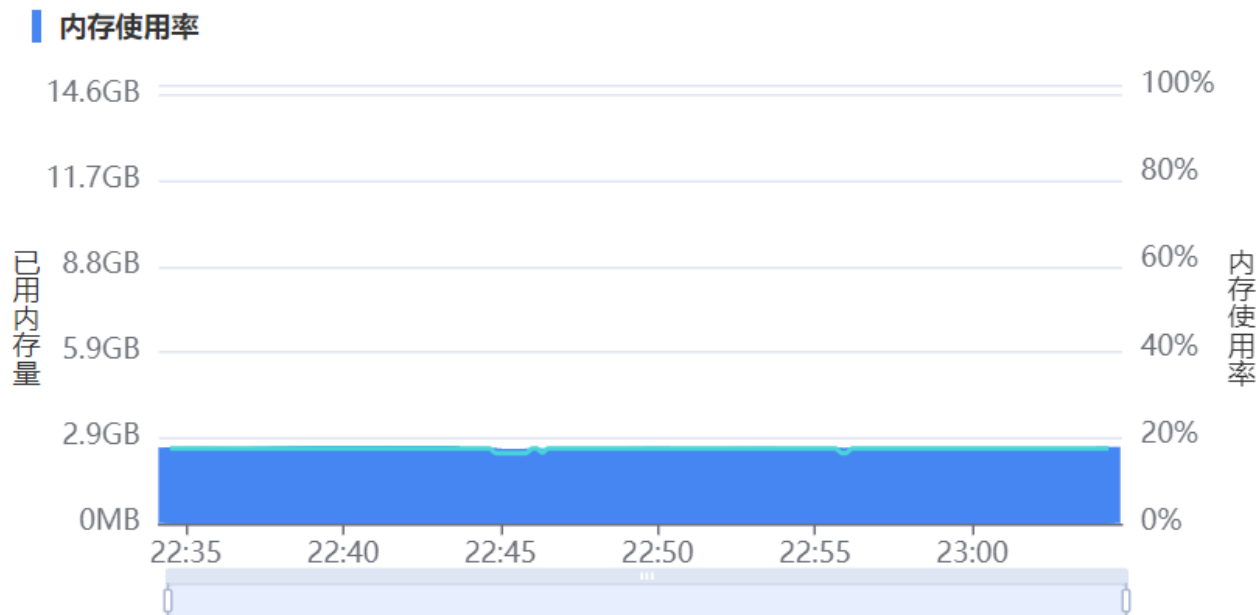
页面刷新于: 2022-01-20 18:12:34 | 自动刷新 ☒ 15 秒

enp2s0f0 实时网络流量分析

进程PID	进程名	发送字节数	接收字节数	总字节数
54790	lightdb	37094	2016	39110
219012	nginx	570	698	1268

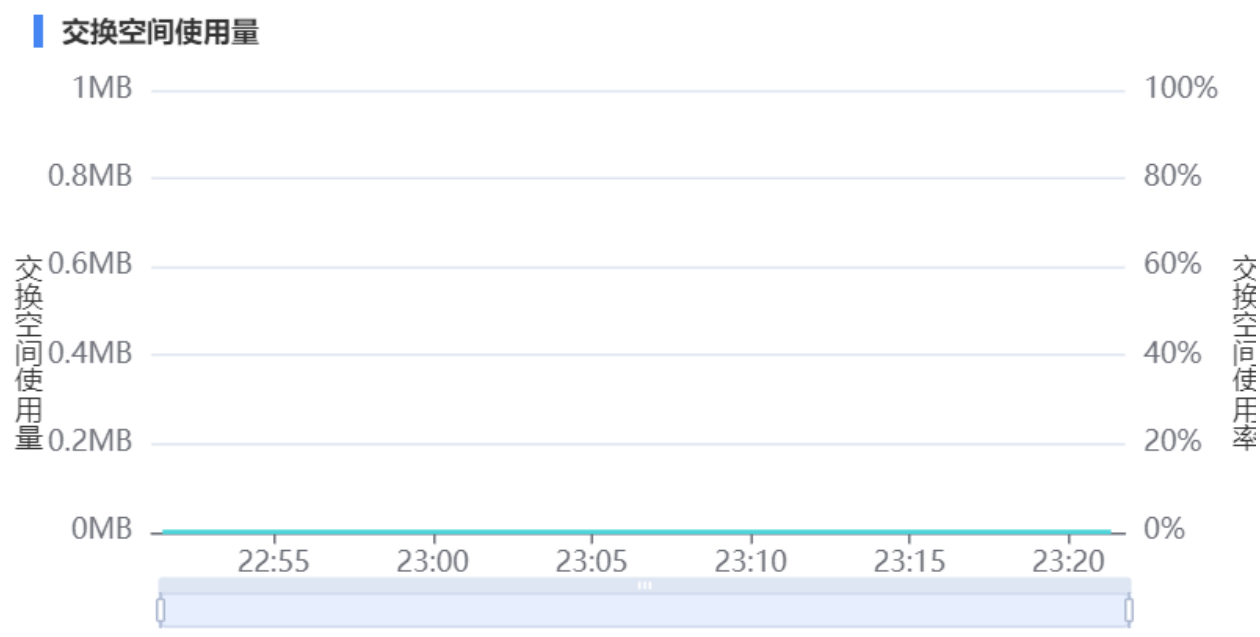
内存使用率

内存使用每 10 秒采集一次，每次展示的是当前采集时间点的系统内存使用情况。



交换空间使用量

交换空间使用每 10 秒采集一次，每次展示的是当前采集时间点的系统交换空间使用情况。



文件系统

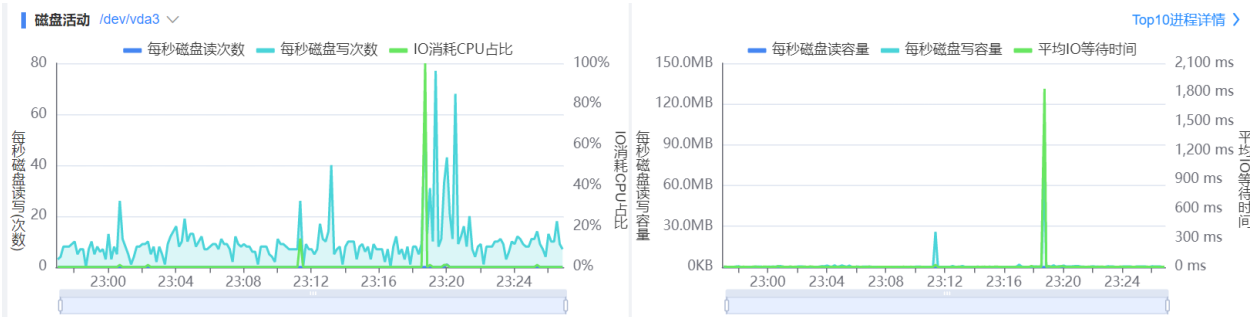
文件系统展示最新的文件系统名称、类型、挂载点和最新的使用情况。

文件系统

设备名称	文件系统	类型	挂载点	总空间	可用空间	使用率(%)
/dev/vda	/dev/vda1	xfs	/	300G	272.3G	9.22

磁盘活动

磁盘活动展示磁盘每秒读写次数、每秒读写容量、IO 消耗 CPU 占比和平均 IO 等待时间



同时展示 Top10 进程详情

Top10进程 (按%util排序)

页面刷新于: 2022-01-23 23:26:56 自动刷新

PID	进程名称	读取速度(KB/s)	写入速度(KB/s)	IO占用(%)
2	kthreadd	0.00	0.00	0.00
4	kworker/0:0H	0.00	0.00	0.00
6	mm_percpu_wq	0.00	0.00	0.00
7	ksoftirqd/0	0.00	0.00	0.00
8	rcu_sched	0.00	0.00	0.00
9	rcu_bh	0.00	0.00	0.00
10	migration/0	0.00	0.00	0.00
11	watchdog/0	0.00	0.00	0.00
12	cpuhp/0	0.00	0.00	0.00
13	cpuhp/1	0.00	0.00	0.00

性能测试

此处生成磁盘性能测试报告：

性能测试

下载性能报告

重新测试

1 second per test

O_DIRECT supported on this platform for open_datasync and open_sync.

Compare file sync methods using one 8kB write:
(in wal_sync_method preference order, except fdatsync is Linux's default)
open_datasync 1237.102 ops/sec 808 usecs/op
fdatsync 1186.694 ops/sec 843 usecs/op
fsync 1035.138 ops/sec 966 usecs/op
fsync_writethrough n/a
open_sync 1027.339 ops/sec 973 usecs/op

3.9 数据库监控与管理

数据库监控与管理监控数据库的一些核心功能，展示数据库的运行状态，方便用户查看数据库运行是否监控，如果出现问题，用户也可以及时的根据这些功能分析定位数据库存在的问题。数据库监控与管理包括PWR 管理、数据库概览、代理监控状态和 TOP SQL。

PWR 管理

数据库每 10 秒定时生成数据库快照。PWR 管理功能真是所有快照，根据时间范围可以查询快照列表。

首页PWR管理PWR报告

192.168.86.10

页面刷新于：2022-04-04 18:02:57

自动刷新

生成时间

请选择

查询

生成报告

手动创建快照

快照设置

开始快照: 3 2022-04-04 20:30:00+08

结束快照: 3 2022-04-04 20:30:00+08

快照	生成时间	设置开始快照	设置结束快照
1	2022-04-04 20:10:00+08	设置开始快照	设置结束快照
2	2022-04-04 20:20:00+08	设置开始快照	设置结束快照
3	2022-04-04 20:30:00+08	设置开始快照	设置结束快照
4	2022-04-04 20:40:00+08	设置开始快照	设置结束快照
5	2022-04-04 20:50:00+08	设置开始快照	设置结束快照
6	2022-04-04 21:00:00+08	设置开始快照	设置结束快照
7	2022-04-04 21:10:00+08	设置开始快照	设置结束快照
8	2022-04-04 21:20:00+08	设置开始快照	设置结束快照
9	2022-04-04 21:30:00+08	设置开始快照	设置结束快照
10	2022-04-04 21:40:00+08	设置开始快照	设置结束快照
11	2022-04-04 21:50:00+08	设置开始快照	设置结束快照

共 36 条

1

50 条/页

跳至

页

LightDB 云控制台 danji-217

913764 system

首页 主机概览 PWR管理

10.20.30.217 页面刷新于: 2022-07-05 14:36:33 自动刷新

生成时间 请选择 查询 生成报告 手动创建快照 快照设置

开始快照: 1833 2022-07-05 14:00:00+08 结束快照: 1835 2022-07-05 14:20:00+08

快照	生成时间	设置开始快照	设置结束快照
1836	2022-07-05 14:30:00+08	设置开始快照	设置结束快照
1835	2022-07-05 14:20:00+08	设置开始快照	设置结束快照
1834	2022-07-05 14:10:00+08	设置开始快照	设置结束快照
1833	2022-07-05 14:00:00+08	设置开始快照	设置结束快照
1832	2022-07-05 13:50:00+08	设置开始快照	设置结束快照
1831	2022-07-05 13:40:00+08	设置开始快照	设置结束快照
1830	2022-07-05 13:30:00+08	设置开始快照	设置结束快照
1829	2022-07-05 13:20:00+08	设置开始快照	设置结束快照
1828	2022-07-05 13:10:00+08	设置开始快照	设置结束快照
1827	2022-07-05 13:00:00+08	设置开始快照	设置结束快照
1826	2022-07-05 12:50:00+08	设置开始快照	设置结束快照

共 978 条 1 2 3 ... 20 50 条/页 跳至 页

首页

集群管控

集群列表

主机管理

实例列表

集群拓扑图

集群拓扑图

主机监控与管理

主机概览

数据库监控与管理

PWR管理

数据库概览

实时SQL

TOP SQL

用户选择 2 个快照可以生成 PWR 报告

首页 PWR管理 PWR报告

PWR报告 下载报告

起始快照 3 2022-04-04 20:30:00+08 结束快照 3 2022-04-04 20:30:00+08

LightDB profile report

Server Name	Platform	CPUs	Cores	Sockets	Memory (GB)	Swap (GB)
lightdb	Linux x86_64	4	4	4	5.7	3.9

	Snap ID	Snap Time
Begin Snap	3	2022-04-04 20:30:00
End Snap	3	2022-04-04 20:30:00
Elapsed	00:00:00	--
DB Time	00:00:00	--

Cluster Name	Role	Release	Startup Time
	primary	LightDB 13.3-22.1	2022-04-04 20:07:55

Report sections

[Server statistics](#)
[Database statistics](#)
[Statement statistics by database](#)
[Cluster statistics](#)
[Tablespace statistics](#)

用户可以设置快照保存时间期限

快照设置

保留时间

0

^

天

∨

取消

确定

用户可以自己创建快照



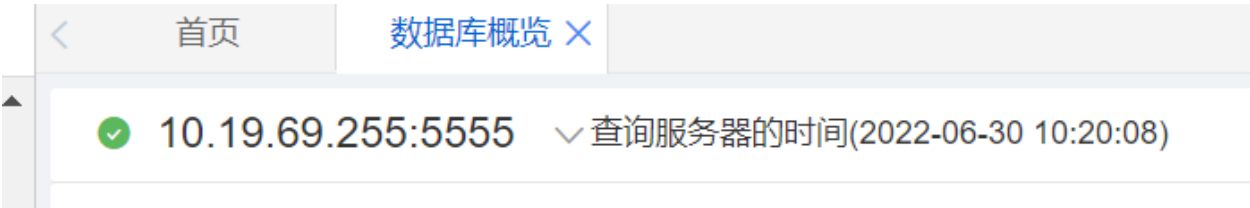
数据库概览

数据库概览展示了数据库核心的配置和运行信息。包括动态修改 GUC 参数、冻结事务比例、节点信息、上次备份、活跃会话数、事务数、缓存命中率、DML 影响行数、等待事件、表膨胀监控、临时文件、WAL 数据量信息和数据库物理写信息、SQL 审核



数据库实例切换

数据库实例切换功能计划针对所有数据库实例进行监控。目前只实现了针对主节点的监控。



时间范围统一选择

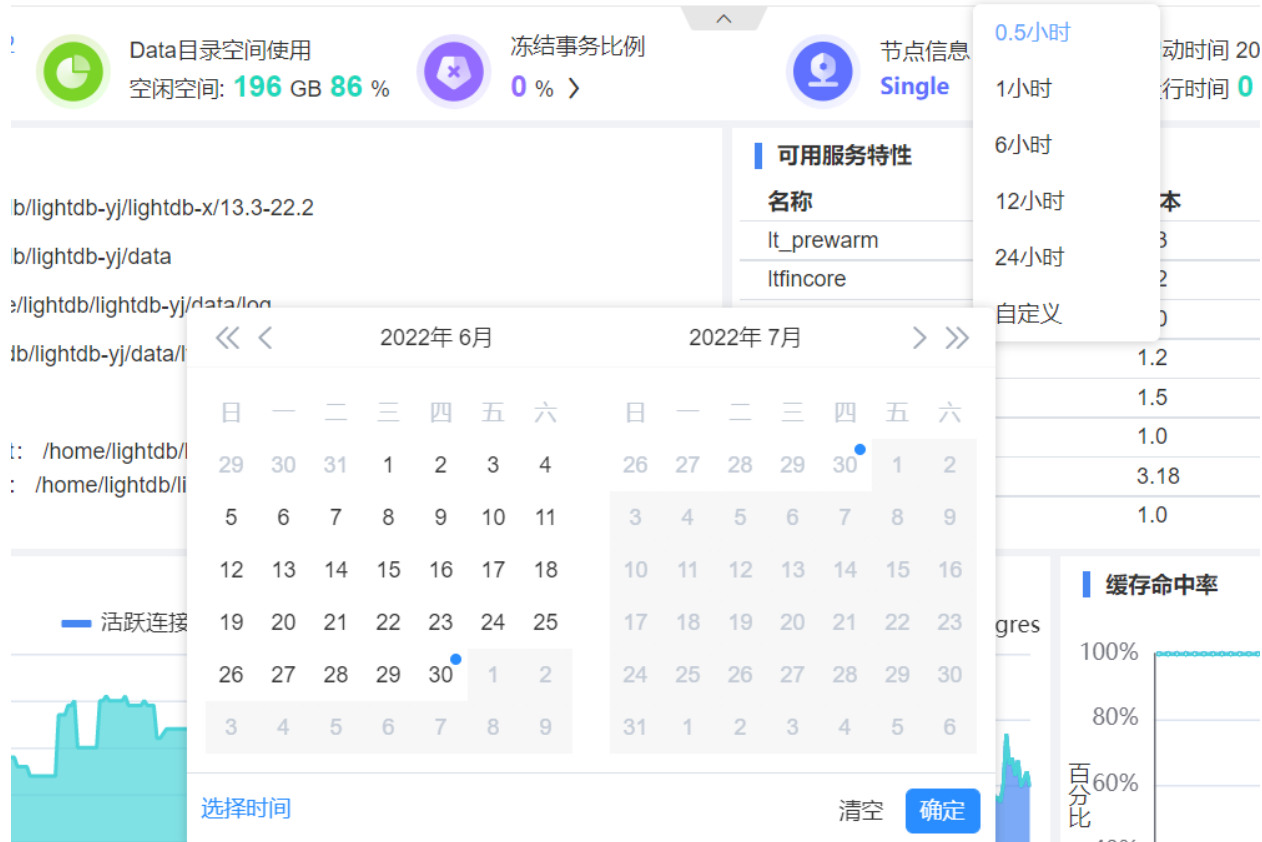
时间范围统一选择提供的选择范围有 0.5 小时、1 小时、6 小时、12 小时、24 小时和自定义范围。通过选择不同的时间范围，页面下面的图表就可以展示不同时间范围的数据，除了自定义时间范围外，其他的时间范围都是以服务器当前时间作为结束时间向前推导开始时间来查询数据。



自定义范围是用户根据实际需要可以选择图表数据范围的起止时间。确定后，页面上展示选择的时间范围，同时图表数据按照选定的时间范围进行加载。首先选择日期

555 查询服务器的时间(2022-06-30 10:30:42)

时间范围 0.5小时 页面刷新于:



日期范围选好后，可以选择时间范围



自动刷新

自动刷新默认是开启的且刷新频率为 5 秒，用户可以根据需要开启。自动刷新频率选择的范围有 1 秒、3 秒、5 秒、10 秒、15 秒、30 秒、1 分钟和 5 分钟。开启自动刷新，选择频率后，页面会根据选择的时间频率加载图表的数据。

Data目录空间使用

空闲空间: 196 GB 86 %

冻结事务比例

0 % >

0.5小时

1小时

6小时

12小时

24小时

自定义

启动时间 2022-06-30 08:57:14

运行时间 0 天 1 小时

lb/lightdb-yj/lightdb-x/13.3-22.2

lb/lightdb-yj/data

lb/lightdb-yj/data/log

开始时间

结束时间

00	00	00	00	00	00
01	01	01	01	01	01
02	02	02	02	02	02
03	03	03	03	03	03
04	04	04	04	04	04
05	05	05	05	05	05
06	06	06	06	06	06
07	07	07	07	07	07
08	08	08	08	08	08

选择日期

清空

确定

版本	所有者
1.3	lightdb
1.2	lightdb
1.0	lightdb
1.2	lightdb
1.5	lightdb
1.0	lightdb
3.18	lightdb
1.0	lightdb

缓存命中率

百分比

42

10.20.31.205:5432

参数名称

当前值

建议值

修改生效类型

更新时间

allow_in_place_tablespaces

off

立即生效

2023-04-04 11:00:00

allow_system_table_mods

off

立即生效

2023-04-04 11:00:00

application_name

lightdbAgent

立即生效

2023-04-04 11:00:00

archive_cleanup_command

立即生效

2023-04-04 11:00:00

archive_command

rm -f /data/lightdb/iv/lightdb...& cp %p /data/lightdb/iv/li...

立即生效

2023-04-04 11:00:00

archive_mode

on

立即生效

2023-04-04 11:00:00

archive_timeout

0

立即生效

2023-04-04 11:00:00

array_nulls

on

立即生效

2023-04-04 11:00:00

authentication_timeout

6

立即生效

2023-04-04 11:00:00

autovacuum

on

立即生效

2023-04-04 11:00:00

autovacuum_analyze_scale_factor

0.

立即生效

2023-04-04 11:00:00

autovacuum_analyze_threshold

50

立即生效

2023-04-04 11:00:00

autovacuum_freeze_max_age

200000000

立即生效

2023-04-04 11:00:00

autovacuum_max_workers

3

立即生效

2023-04-04 11:00:00

autovacuum_multixact_freeze_max_age

400000000

立即生效

2023-04-04 11:00:00

autovacuum_naptime

900

15min

立即生效

2023-04-04 11:00:00

autovacuum_vacuum_cost_delay

2

立即生效

2023-04-04 11:00:00

autovacuum_vacuum_cost_limit

-1

立即生效

2023-04-04 11:00:00

autovacuum_vacuum_insert_scale_factor

0.2

立即生效

2023-04-04 11:00:00

autovacuum_vacuum_insert_threshold

1000

立即生效

2023-04-04 11:00:00

autovacuum_vacuum_scale_factor

0.2

立即生效

2023-04-04 11:00:00

修改GUC参数

参数名称

当前值

建议值

修改生效类型

allow_system_table_mods

off

立即生效

archive_command

rm -f /data/lightdb/...

立即生效

archive_timeout

0

0

立即生效

重置

取消

确定

共 397 条

1

2

3

...

8

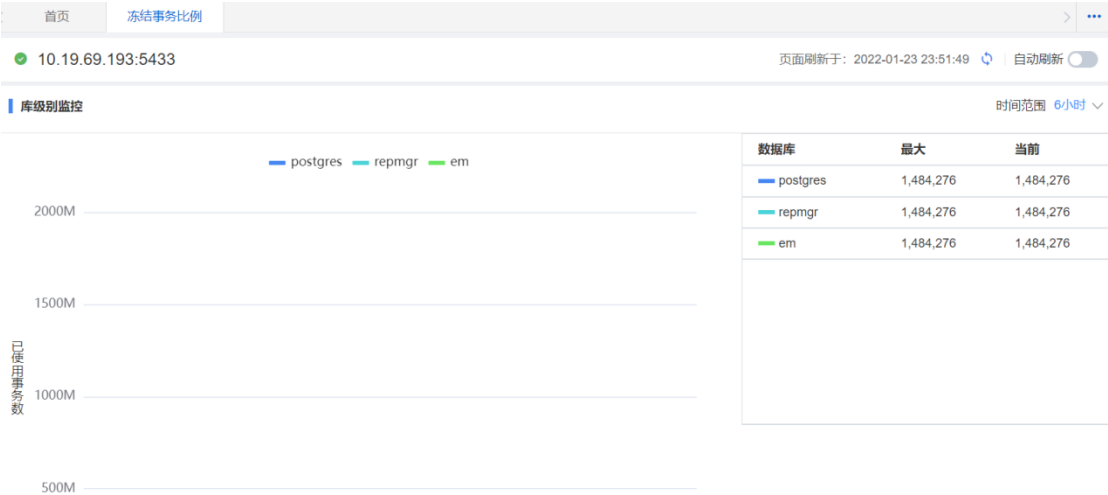
>

50 条/页

跳至

页

冻结事务比例



数据库表年龄信息

点击冻结事务比例中数据库可查看具体表年龄信息，并且可根据表格列名排序

首页实例列表10.20.31.205的数据库概览GLUC参数集群拓扑图主机管理冻结事务表级详情冻结事务表级到详情冻结事务表级到详情...

页面刷新于: 2023-04-04 13:43:07 自动刷新

数据库postgres的表详情

表名	活元组个数	死元组个数	死元组/活元组比例	表年龄	Vacuum次数	Toast名字	Toast年龄	表最大年龄
hint_plan_hints	0	0	0	503655	2	pg_toast_20116	503655	503655
ufl_file ufl_file_dir	0	0	0	503653	2	pg_toast_20426	503653	503653
lt_catalog_deps_saved_ddl	0	0	0	503653	2	pg_toast_22386	503653	503653
zhparser.zhprs_custom_word	0	0	0	503575	2	pg_toast_22430	503575	503575
lt_catalog_sample_activity_history	1284	1059	0.8248	503568	87	pg_toast_23340	503568	503568
lt_catalog_sample_activity_profile (5.0757575757575% bloom)	82375	0	0	503568	11	pg_toast_23351	503568	503568
lt_catalog_sample_active_session_history	633	521	0.8231	503568	87			503568
lt_catalog_sample_active_session_profile	1301	0	0	503568	2			503568
lt_catalog lt_sm_key_stat	0	0	0	503567	2	pg_toast_23412	503567	503567
lt_catalog lt_sm_key_ext_stat	0	0	0	503567	2	pg_toast_23425	503567	503567
cron.job_run_details	28203	0	0	503564	2	pg_toast_23694	503564	503564
cron.job	5	0	0	503564	2	pg_toast_23673	503564	503564
cron.lt_job_ext	5	0	0	503564	2	pg_toast_23708	503564	503564
lt_catalog_sample_statements_total	391	0	0	503562	2	pg_toast_23925	503562	503562
lt_catalog_sample_wait_event_total	82133	0	0	503562	10	pg_toast_23963	503562	503562
lt_catalog tablespaces_list	2	12	6	503562	2	pg_toast_23969	503562	503562
lt_catalog sample_stat_tablespaces	260	0	0	503562	2			503562
lt_catalog last_stat_tablespaces	2	2	1	503562	6	pg_toast_24001	503562	503562
lt_catalog sample_stat_tables	10356	0	0	503562	6			503562
lt_catalog last_stat_tables	476	476	1	503562	85			503562
lt_catalog sample_stat_tables_total	1560	0	0	503562	2			503562
lt_catalog indexes_list	120	0	0	503562	2			503562

主节点信息

主节点信息展示数据库集群信息。数据库集群通过 Itcluster 来实现，集群状态通过执行 Itcluster 命令来查询 Itcluster -f \$Itcluster_config_path cluster show

首页集群监控...

10.19.69.193:5433 页面刷新于: 2022-01-23 23:52:25 自动刷新

集群监控

当前节点: 主节点 备节点数量: 1

IP	端口	角色	实例状态	守护进程状态	复制模式	查看详情
10.19.69.193	5433	primary	* running	running		查看详情
10.19.69.192	5433	standby	running	running	sync	

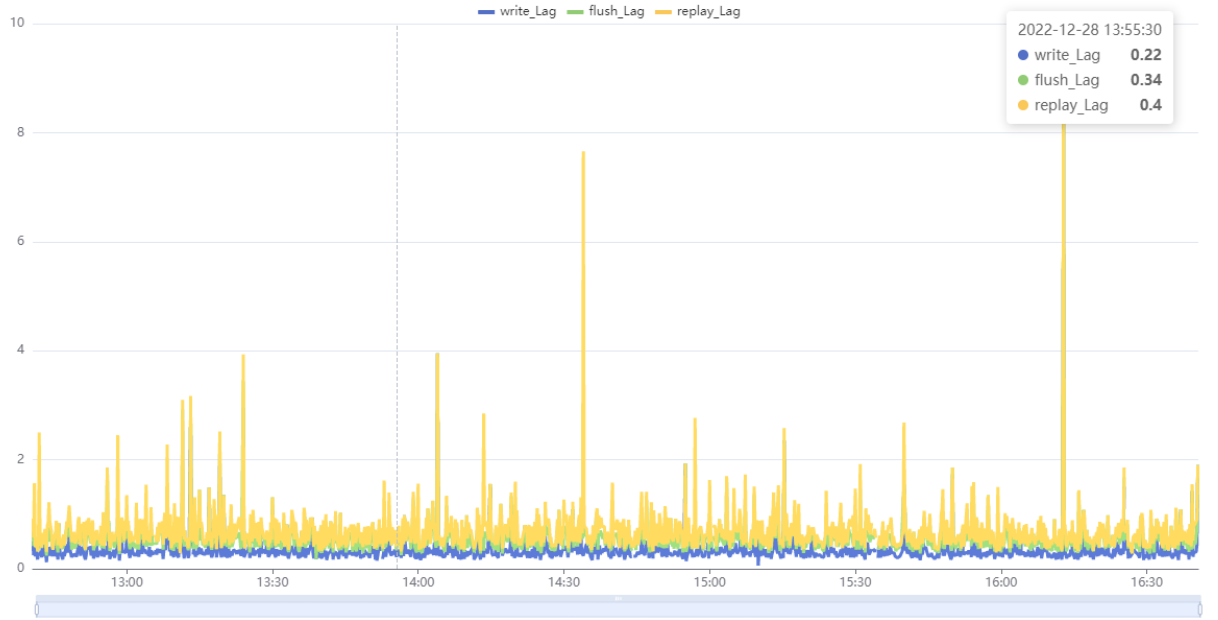
通过查看详情，可以查看集群主节点复制延迟和 LSN 状态

集群监控

当前节点: 主节点 备节点数量: 1

备节点 10.20.31.207:5432

- write_lag : replication time delay written to disk by this standby(ms)
- flush_lag : replication time delay flushed to disk by this standby(ms)
- replay_lag : replication time delay applied into the database on this standby server(ms)



首页 集群监控 集群监控详情

集群监控

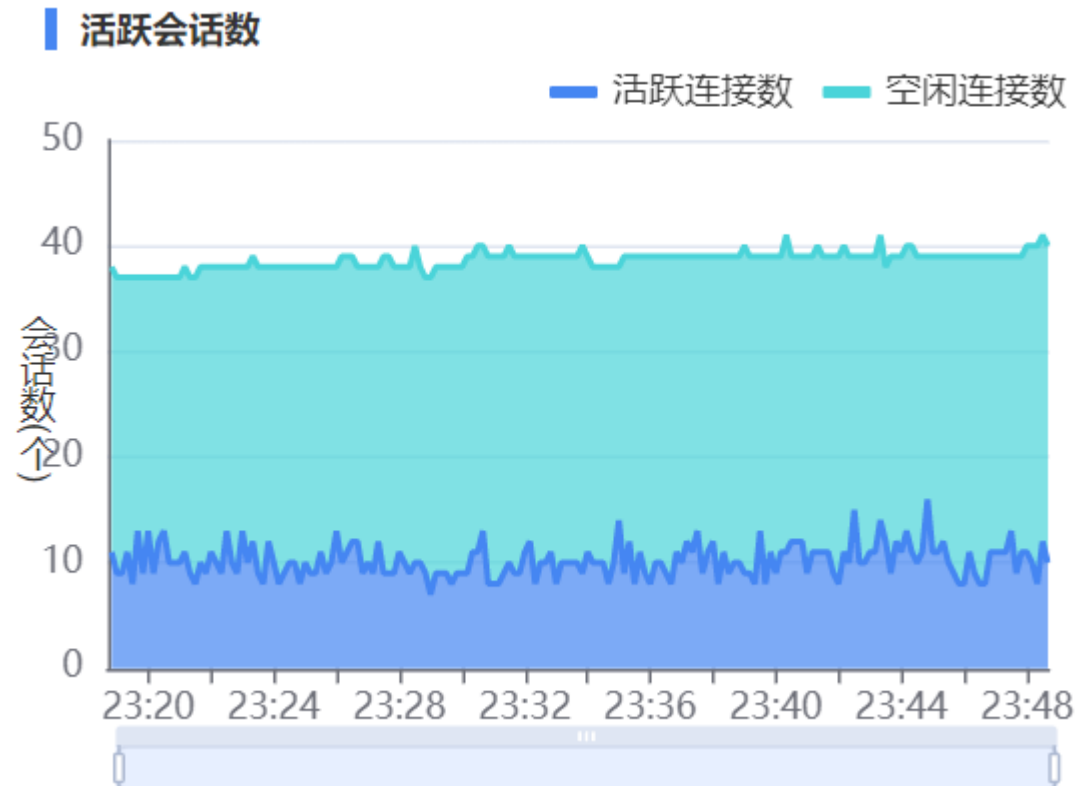
当前节点: 主节点 备节点数量: 1

复制延迟(ms)	复制延迟(bytes)	LSN状态		
instancelp	sampleTime	writesn	flushsn	replaysn
10.19.36.9	2022-01-30 14:42:00	A/2A4F5758(709,842,776)	A/2A4F5758(709,842,776)	A/2A4F5758(709,842,776)
10.19.36.9	2022-01-30 14:41:50	A/2A4BAB68(709,602,152)	A/2A4BAB68(709,602,152)	A/2A4BAB68(709,602,152)
10.19.36.9	2022-01-30 14:41:40	A/2A4452C0(709,120,704)	A/2A4452C0(709,120,704)	A/2A4452C0(709,120,704)
10.19.36.9	2022-01-30 14:41:30	A/2A377B70(708,279,152)	A/2A377B70(708,279,152)	A/2A377B70(708,279,152)
10.19.36.9	2022-01-30 14:41:20	A/2A331D88(707,992,968)	A/2A331D88(707,992,968)	A/2A331D88(707,992,968)
10.19.36.9	2022-01-30 14:41:10	A/2A2DE1A0(707,649,952)	A/2A2DE1A0(707,649,952)	A/2A2DE1A0(707,649,952)
10.19.36.9	2022-01-30 14:41:00	A/2A1A0FA8(706,351,016)	A/2A1A0FA8(706,351,016)	A/2A1A0FA8(706,351,016)
10.19.36.9	2022-01-30 14:40:50	A/2A144478(705,971,320)	A/2A144478(705,971,320)	A/2A144478(705,971,320)
10.19.36.9	2022-01-30 14:40:40	A/2A0F6298(705,651,352)	A/2A0F6298(705,651,352)	A/2A0F6298(705,651,352)

共 39034 条 1 2 3 ... 781 50 条/页 跳至 页

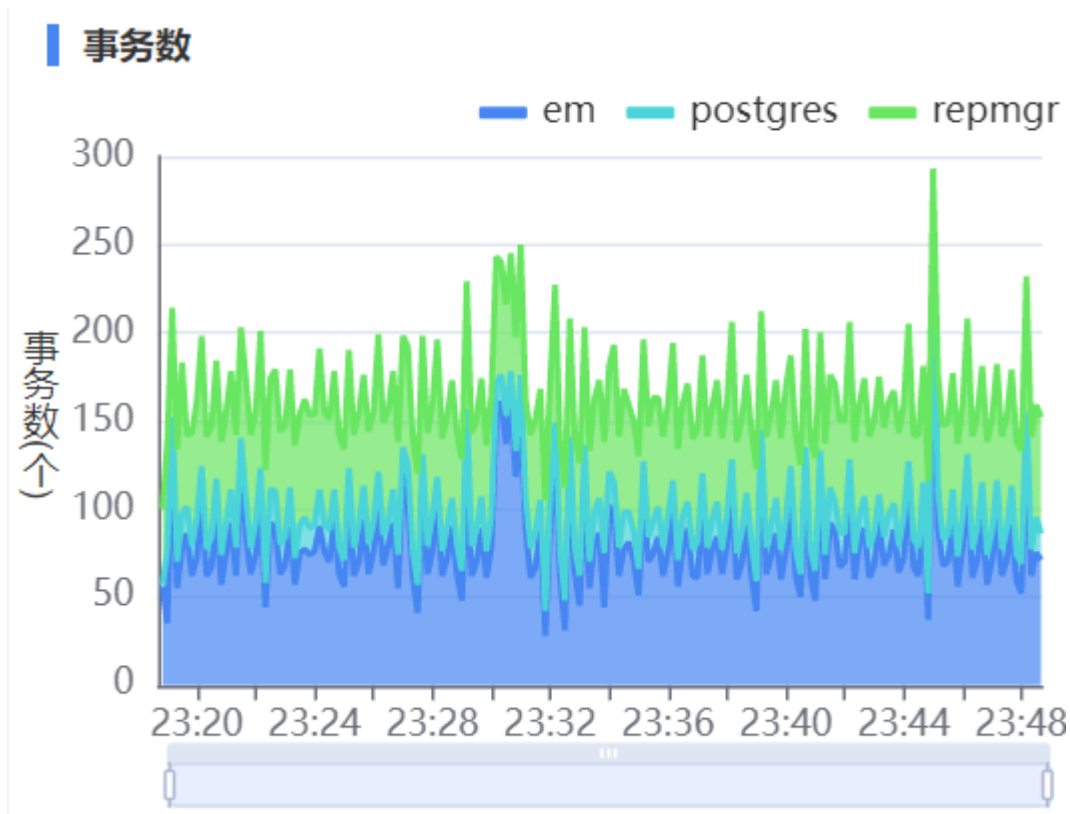
上次备份

本实例在集中备份模块最后一次执行备份的时间。活跃会话数 “ ” 活跃会话数统计视图
lt_stat_activity 总连接数，然后根据条件 state like ' idle%' 查询空闲连接数，根据总连接数-空闲连接数 = 活跃
连接数



事务数

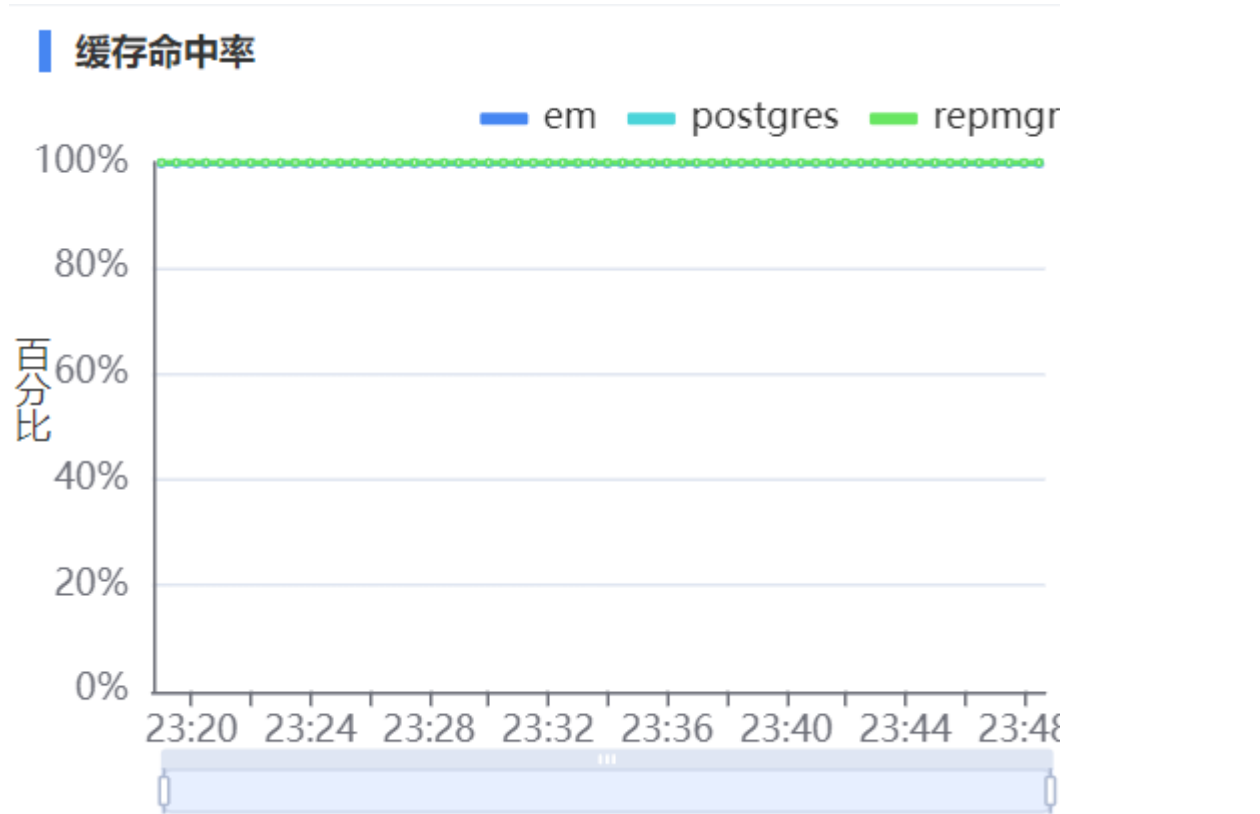
数据库事务数从视图 pg_stat_database 中获取 xact_commit 的值，xact_commit 代表此数据库中已提交的事务数。



缓存命中率

缓存命中率从视图 `pg_statio_all_tables` 中数据。`pg_statio_all_table` 视图中当前数据库每个表一行 (包括 TOAST 表), 显示: 表 OID、模式名、表名、从该表中读取的磁盘块总数、缓冲区命中次数、该表上所有索引的磁盘块读取总数、该表上所有索引的缓冲区命中总数、在该表的辅助 TOAST 表 (如果存在) 上的磁盘块读取总数、在该表的辅助 TOAST 表 (如果存在) 上的缓冲区命中总数、TOAST 表的索引的磁盘块读取总数、TOAST 表的索引的缓冲区命中总数。当前数据库中的每个表一行, 显示有关在指定表上 I/O 的统计信息。缓存命中率图表展示总的命中率数据, 计算公式:

```
sum(COALESCE(t.heap_blks_hit,0)
+COALESCE(t.idx_blks_hit,0)
+COALESCE(t.toast_blks_hit,0)
+COALESCE(t.tidx_blks_hit,0)) over() *100
/
sum(COALESCE(t.heap_blks_read,0)
+COALESCE(t.heap_blks_hit,0)
+COALESCE(t.idx_blks_read,0)
+COALESCE(t.idx_blks_hit,0)
+COALESCE(t.toast_blks_read,0)
+COALESCE(t.toast_blks_hit,0)
+COALESCE(t.tidx_blks_read,0)
+COALESCE(t.tidx_blks_hit,0)) over() as decimal(10,2)
```



点击命中率图表上的每个点，可以查看缓存命中率明细：

repmgr

模式名称 表名称 查询 重置

请输入列表标题名称

共 44 条

模式名称	表名称	缓存命中率	当前表读取磁	表缓冲区命中	表索引读取磁	表索引缓冲区	表TOAST表读	表TOAST表写	表TOAST表删	表TOAST表查	采集时间点
repmgr	events	72.73	3	8	0	0	0	0	0	0	2022-01-30 10:00:00
pg_catalog	lt_class_ext	100	6	175161	6	350314	0	0	0	0	2022-01-30 10:00:00
repmgr	monitoring_	99.46	2365	259954	886	338290	0	0	0	0	2022-01-30 10:00:00
repmgr	nodes	100	3	338948	6	262442	0	0	0	0	2022-01-30 10:00:00
pg_catalog	pg_aggregat	99.79	4	1569	6	3139	0	0	0	0	2022-01-30 10:00:00
pg_catalog	pg_am	100	3	900239	0	0	0	0	0	0	2022-01-30 10:00:00
pg_catalog	pg_amop	100	21	3276985	25	3545464	0	0	0	0	2022-01-30 10:00:00
pg_catalog	pg_amproc	100	11	1703579	9	4130691	0	0	0	0	2022-01-30 10:00:00
pg_catalog	pg_attrdef	100	3	189198	8	365918	0	0	0	0	2022-01-30 10:00:00
pg_catalog	pg_attribute	100	155	10593913	40	21533782	0	0	0	0	2022-01-30 10:00:00

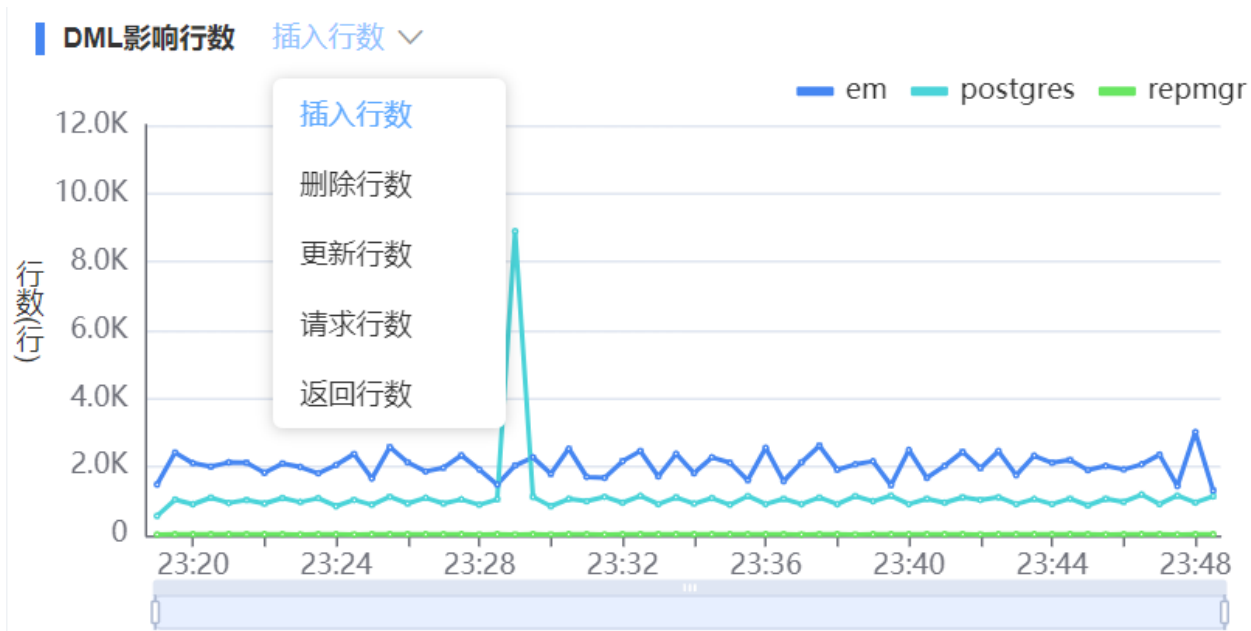
共 44 条 1 50 条/页 跳至 页

pg_statio_all_tables 视图列说明：

列名	类型	描述
relid	oid	表的 OID
schemaname	name	该表所在的模式名
relname	name	这个表的名称
heap_blks_read	bigint	从该表中读取的磁盘块的数量
heap_blks_hit	bigint	该表中的缓冲区命中数
idx_blks_read	bigint	从这个表上所有索引读取的磁盘块数
idx_blks_hit	bigint	这个表上所有索引中的缓冲区命中数
toast_blks_read	bigint	从这个表的 TOAST 表中读取的磁盘块的数量(如果有的话)
toast_blks_hit	bigint	这个表的 TOAST 表中的缓冲区命中数(如果有的话)
tidx_blks_read	bigint	从这个表的 TOAST 表索引中读取的磁盘块的数量(如果有的话)
tidx_blks_hit	bigint	这个表的 TOAST 表索引中的缓冲区命中数(如果有的话)

DML 影响行数

DML 影响行数从视图 `pg_stat_all_tables` 中数据。`pg_stat_all_tables` 视图将为当前数据库中的每一个表（包括 TOAST 表）包含一行，该行显示与对该表的访问相关的统计信息。



点击 DML 影响行数图表上的每个点，可以查看 DML 影响行数明细：

首页	数据库概览	DML影响行数	
----	-------	---------	--

postgres

模式名称 请输入 表名称 请输入 查询 重置

请输入列表标题名称 共 211 条 1 2 3 ... 5 >

模式名称	表名称	Insert 行数	Update 行数	Delete 行数	Fetch 行数	采集时间点
public	baselines	0	0	0	0	2022-01-30 23:09:00
public	bl_samples	0	0	0	0	2022-01-30 23:09:00
columnar	chunk	0	0	0	0	2022-01-30 23:09:00
columnar	chunk_group	0	0	0	0	2022-01-30 23:09:00
public	funcs_list	0	0	0	0	2022-01-30 23:09:00
hint_plan	hints	0	0	0	0	2022-01-30 23:09:00
public	import_queries	0	0	0	0	2022-01-30 23:09:00
public	import_queries_version	0	0	0	0	2022-01-30 23:09:00
public	indexes_list	0	0	0	464	2022-01-30 23:09:00
cron	job	0	0	0	8	2022-01-30 23:09:00

共 211 条 1 2 3 ... 5 > 50 条/页 跳至 页

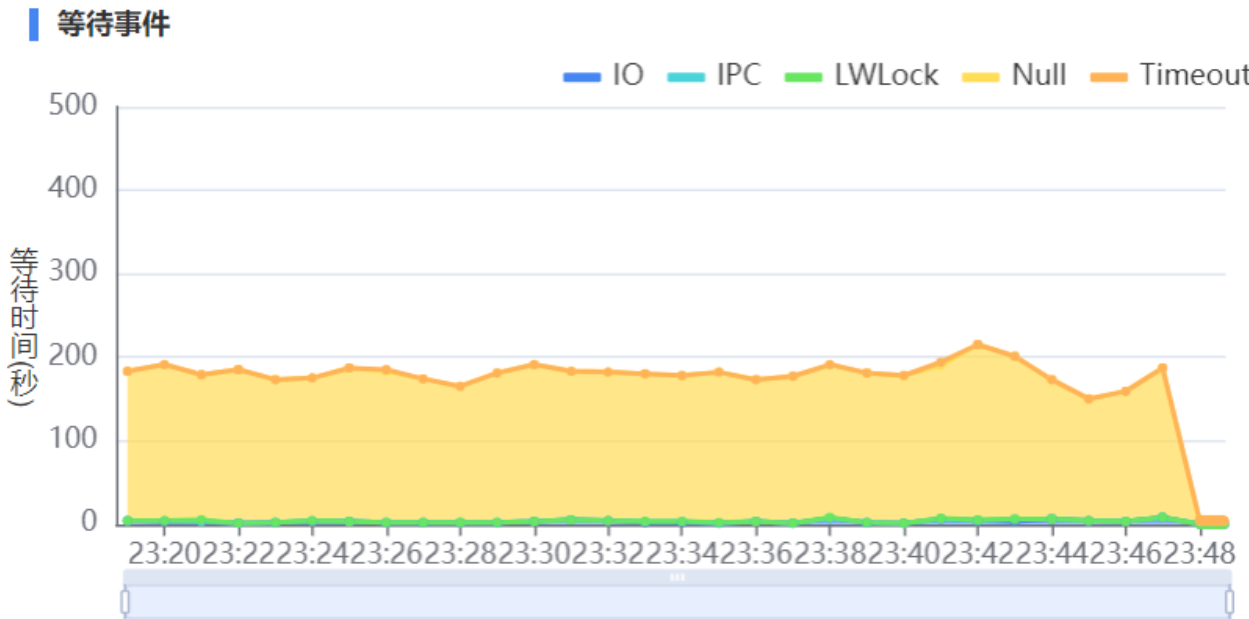
pg_stat_all_tables 视图列说明：

列名	类型	描述
relid	oid	表的 OID
schemaname	name	该表所在的模式的名称
relname	name	这个表的名称
seq_scan	bigint	在此表上启动的顺序扫描数
seq_tup_read	bigint	连续扫描获取的实时行数
idx_scan	bigint	对这个表发起的索引扫描数
idx_tup_fetch	bigint	索引扫描获取的实时行数
n_tup_ins	bigint	插入的行数
n_tup_upd	bigint	更新的行数(包括 HOT 更新的行)
n_tup_del	bigint	删除的行数
n_tup_hot_upd	bigint	HOT 更新的行数(即，不需要单独的索引更新)
n_live_tup	bigint	活的行的估计数量
n_dead_tup	bigint	僵死行的估计数量
n_mod_since_analyze	bigint	自上次分析此表以来修改的行的估计数量
n_ins_since_vacuum	bigint	自上次清空此表以来插入的行的估计数量
last_vacuum	timestamp with time zone	最后一次手动清理这个表(不包括 VACUUM FULL)
last_autovacuum	timestamp with time zone	这个表最后一次被自动清理守护进程清理的时间
last_analyze	timestamp with time zone	上一次手动分析这个表
last_autoanalyze	timestamp with time zone	自动清理守护进程最后一次分析这个表
vacuum_count	bigint	这个表被手动清理的次数(VACUUM FULL 不计数)
autovacuum_count	bigint	这个表被 autovacuum 守护进程清理的次数
analyze_count	bigint	手动分析这个表的次数
autoanalyze_count	bigint	这个表被 autovacuum 守护进程分析的次数

等待事件

DML 影响行数

等待事件从表 sample_activity_profile 关联 sample_activity_history 表获取数据



通过等待事件图表上的点可以访问等待事件明细：

等待事件

请输入

查询

重置

等待事件类别	等待事件	等待时长/秒	SQL详情	采集时间
Null	Null	33	select count(*) from cron.job_run_details	2022-01-30 23:17:00
Null	Null	33		2022-01-30 23:17:00
Null	Null	33	SELECT collect_activity_history()	2022-01-30 23:17:00
Null	Null	33	SELECT take_sample()	2022-01-30 23:17:00
Null	Null	1	SELECT collect_activity_profile()	2022-01-30 23:17:00
Null	Null	1	select jrd.jobid,jrd.runid,jrd.job_pid as jobPid,jrd.database,jrd.u	2022-01-30 23:17:00
Null		134	汇总	2022-01-30 23:17:00

等待事件类型：

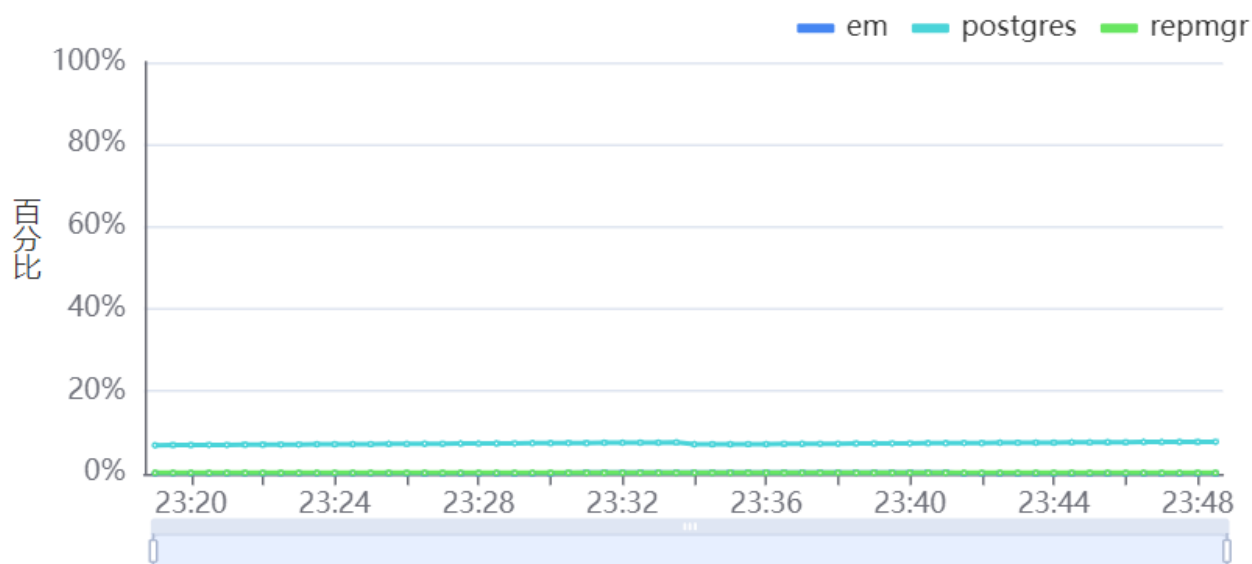
Activity	服务器进程空闲。此事件类型表示在其主处理循环中等待活动的进程。wait_event 将识别特定的等待点。
BufferPin	服务器进程正在等待对数据缓冲的独占访问。 如果另一个进程持有一个打开的游标，该游标最后一次从相关缓冲区读取数据，则缓冲区锁等待可能是漫长的。
Client	服务器进程正在等待连接到用户应用程序的套接字上的活动。 因此，服务器预计发生一些独立于其内部进程的事情。wait_event 将识别特定的等待点。
Extension	服务器进程正在等待扩展模块定义的某个条件。
IO	服务器进程正在等待一个 I/O 操作完成。wait_event 将识别特定的等待点。
IPC	服务器进程正在等待与另一个服务器进程进行交互。wait_event 将识别特定的等待点。
Lock	服务器进程正在等待一个重量级锁。重量级锁，也称为锁管理器锁或简单锁，主要保护表等 SQL 可见对象。然而，它们也用于确保某些内部操作的互斥，例如关系扩展。wait_event 将识别等待的锁的类型。
LWLock	服务器进程正在等待一个轻量级锁。大多数这样的锁保护共享内存中的特定数据结构。wait_event 将包含标识轻量级锁用途的名称。(有些锁有特定的名称；其他锁是一组锁的一部分，每个锁具有类似的目的。)
Timeout	服务器进程正在等待超时过期。wait_event 将识别特定的等待点。

表膨胀监控

表膨胀也是从视图 pg_stat_all_tables 中获取数据计算出来结果。表膨胀采集逻辑较为苛刻，页面没数据是正常的情况。获取表膨胀逻辑：

```
with tmp as(select schemaname,relname as tablename,pg_relation_size(''||schemaname||'.'.
→'||relname||') tab_size,
    n_dead_tup,
    n_live_tup,
    coalesce(round(n_dead_tup * 100 / (case when n_live_tup + n_dead_tup = 0 then null
→else n_live_tup + n_dead_tup end ),2),0.00) as dead_tup_ratio,
    round( case when (sum(n_live_tup + n_dead_tup) over())=0 then 0
            else (sum( n_dead_tup) over())*100/(sum(n_live_tup + n_dead_tup)
→over()) end ,2) dead_tup_ratio_total
from pg_stat_all_tables)
SELECT * FROM tmp WHERE schemaname not in ('pg_catalog','lt_catalog')
and tab_size > 100*1024*1024
and n_dead_tup > 1000
and n_live_tup > 1000
and dead_tup_ratio > 20
ORDER BY dead_tup_ratio DESC LIMIT 1000;
```

表膨胀监控



通过表膨胀监控图表上的点可以访问表膨胀明细：

postgres

模式名称 表名称 [查询](#) [重置](#)

请输入列表标题名称

共 211 条

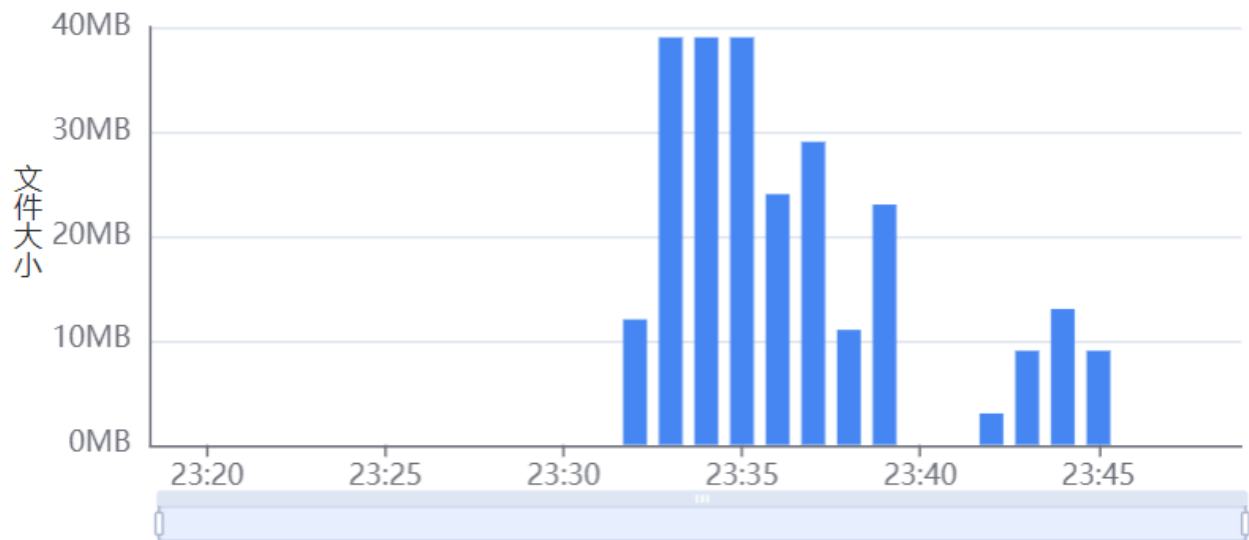
模式名称	表名称	表大小/字节	存活元组数量	死元组数量	膨胀率	采集时间点
public	baselines	0	0	0	0	2022-01-30 23:25:30
public	bl_samples	0	0	0	0	2022-01-30 23:25:30
columnar	chunk	0	0	0	0	2022-01-30 23:25:30
columnar	chunk_group	0	0	0	0	2022-01-30 23:25:30
public	funcs_list	0	0	0	0	2022-01-30 23:25:30
hint_plan	hints	0	0	0	0	2022-01-30 23:25:30
public	import_queries	49152	41	0	0	2022-01-30 23:25:30
public	import_queries_version	8192	4	0	0	2022-01-30 23:25:30
public	indexes_list	32768	140	27	16	2022-01-30 23:25:30
cron	job	8192	4	4	50	2022-01-30 23:25:30

共 211 条

临时文件

数据库临时文件信息是从数据库日志中解析出来的数据。系统会把数据库日志定时采集到 lem_db_temp_file_log 表中，通过解析 lem_db_temp_file_log 中的数据即得到临时文件信息。临时文件数据获取逻辑：select db_log_time_minute as minuteTime , round(sum(temp_file_size) /(1024*1024),2) as size from lem_db_temp_file_log group by db_log_time_minute order by 1 asc

临时文件



通过点击临时文件图表中的柱子，可以查看临时文件日志明细：

首页

数据库概览

临时文件 ×

10.19.36.10:5555

页面刷新于: 2022-01-30 23:59:29 | 自动刷新

临时文件	文件大小(MB)	时间	sql
"base/pgsql_tmp/pgsql_tmp464341.474175"	0.86	2022-01-25 14:38:26 CST	SQL statement "delete from cron.job_run_details where runid in (select runid from cron.job_run_details order by runid desc limit (select count(runid) from cron.job_run_details) offset 100000)"
"base/pgsql_tmp/pgsql_tmp464341.474180"	0.86	2022-01-25 14:38:26 CST	SQL statement "delete from cron.job_run_details where runid in (select runid from cron.job_run_details order by runid desc limit (select count(runid) from cron.job_run_details) offset 100000)"
"base/pgsql_tmp/pgsql_tmp464341.474172"	0.86	2022-01-25 14:38:26 CST	SQL statement "delete from cron.job_run_details where runid in (select runid from cron.job_run_details order by runid desc limit (select count(runid) from cron.job_run_details) offset 100000)"
"base/pgsql_tmp/pgsql_tmp464341.47417"	0.85	2022-01-25 14:38:26 CST	SQL statement "delete from cron.job_run_details where runid in (select runid from cron.job_run_details order by runid desc limit (select count(runid) from cron.job_run_details) offset 100000)"

共 150 条

< 1 2 3 ... 15 >

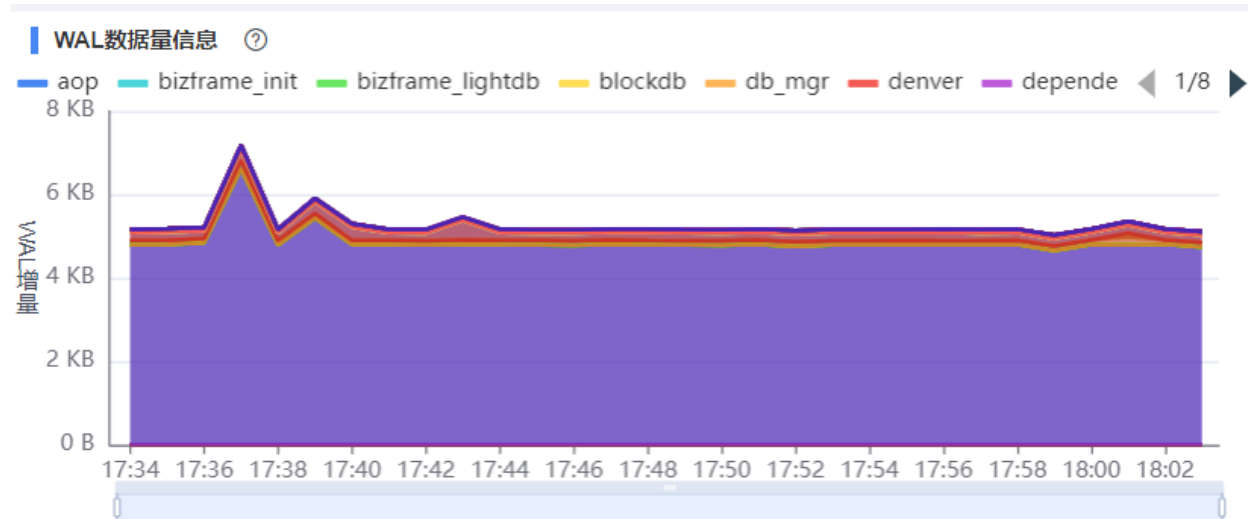
10 条/页

跳至

WAL 数据量信息

WAL 数据量信息从视图 `pg_stat_statements` 中获取数据。WAL 汇总数据获取逻辑：

```
select pd.datname as dbName, sum(pss.wal_bytes) as walAmount
from Pg_stat_statements pss
join pg_database pd on pss.dbid = pd.oid
where pss.wal_bytes > 0 and datname != 'template1'
group by pd.datname
order by datname;
```



通过 WAL 数据量信息图表上的点可以访问 WAL 增量明细：

WAL详情

repmgr

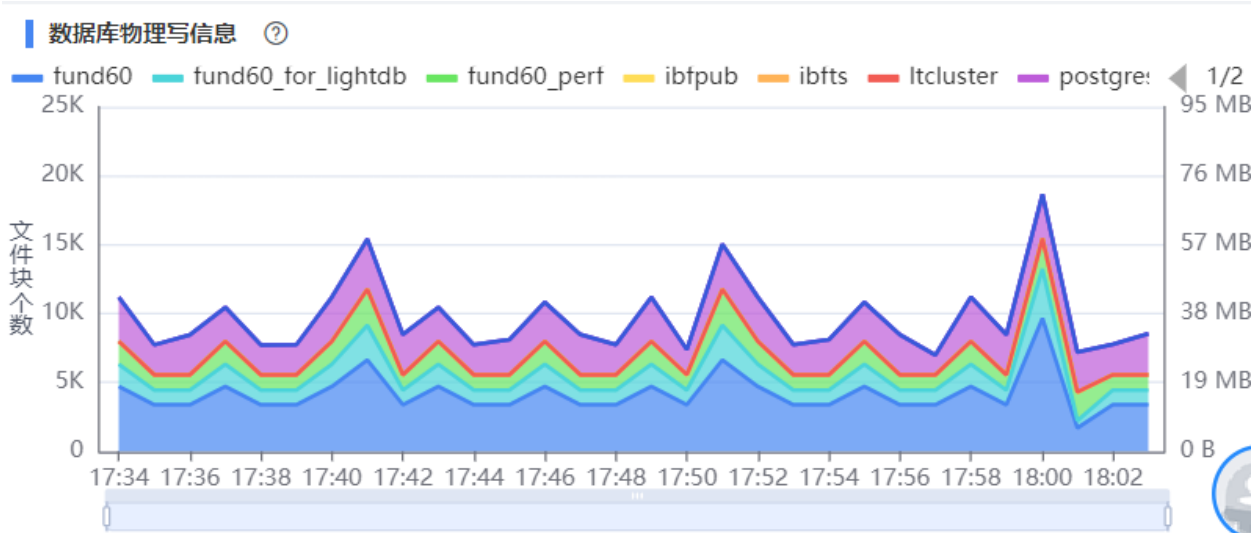
共 1 条

数据库名称	WAL日志数据量(字节)	SQL详情	采集时间
repmgr	38339	INSERT INTO repmgr.monitoring_history (primary_node_id, standby_node_id, last_mon	2022-01-23 23:36:00

共 1 条 < 1 > 50 条/页 跳至 页

数据库物理写信息

数据库物理写从视图 pg_stat_statements 中获取数据。从节点无物理写信息。



通过数据库物理写信息图表上的点可以访问物理写明细：

数据库物理写信息

repmgr

共 1 条

数据库名称	写入共享块总数	写入本地块总数	写入临时块总数	写入总块数	SQL详情	采集时间
repmgr	3	0	0	3	INSERT INTO repmgr.monitoring_history (primary_node_id, standby_node_id, ...)	2022-01-23 23:...

共 1 条 50 条/页 跳至 页

实时 SQL

实时 SQL 展示实时活动的 sql。通过表 lt_stat_activity 得到实时 SQL，得到的数据中文化展示。

10.19.69.255:5555

页面更新于: 2022-06-30 13:32:42

自动刷新

共 7 条

1

	Pid	LeaderPid	用户名称	数据库名称	应用名称	客户端地址	客户端主机名	客户端端口	状态	状态最后一次更新时间	等待事件类型	等待事件	查询开始时间	QueryId	查询语句
☐	29106		lightdb	em	lightdbem	10.20.31.205		58520	idle in transact	2022-06-30 13	Client	ClientRead	2022-06-30 13		select host_id from lem_host where host,
☐	26623		lightdb	em	lightdbem	10.188.131.245		53214	idle in transact	2022-06-30 13	Client	ClientRead	2022-06-30 13		select host_id from lem_host where host,
☐	22677		lightdb	em	lightdbem	10.188.132.12		58374	idle in transact	2022-06-30 13	Client	ClientRead	2022-06-30 13		select cron_id,host_id,host_ip,cron_key,cr
☐	27518		lightdb	em	lightdbem	10.188.131.245		53246	idle in transact	2022-06-30 13	Client	ClientRead	2022-06-30 13		select host_id from lem_host where host,
☐	14480		lightdb	em	lightdbem	10.188.131.245		58366	idle in transact	2022-06-30 13	Client	ClientRead	2022-06-30 13		select host_id from lem_host where host,
☒	27687		lightdb	postgres	lightdbem	10.19.69.255		45598	active	2022-06-30 13			2022-06-30 13		SELECT count(0) FROM lt_stat_activity AS

实时 SQL 搜索条件数据也是中文化展示。

10.19.69.255:5555

页面更新于: 2022-06-30 13:32:42

自动刷新

1

用户名称

请输入

QueryId

请输入

Pid

请输入

应用名称

请输入

状态

请选择

Xact启动时间范围

选择日期

查询启动时间范围

选择日期

客户端地址

请输入

查询

重置

锁详情页面

LightDB 云控制台

dev

10.19.69.255:5555

页面更新于: 2022-06-28 16:35:17

自动刷新

共 2 条

1

	Detail	用户名称	应用名称	QueryId	数据库名称	行数	查询步数	Pid	LeaderPid	应用名称	客户端地址	客户端主机名	客户端端口	Backend开始时间	Xact开始时间	Query开始时间	状态
☐		lightdb			postgres	4	4	51602		lightdbem	10.20.31.205		58748	2022-06-28 10	2022-06-28 16	2022-06-28 16	2022-
☒		lightdb			postgres	4	4	102189		lightdbem	10.20.31.205		52306	2022-06-28 13	2022-06-28 16	2022-06-28 16	2022-

实时 SQL 表和数据项说明：

对象。	类型。	说明。
lt_stat_activity。	视图。	每个服务器进程将有一行，显示与该进程当前活动相关的信息。
username。	数据项。	登录到这个后端的用户的 OID。
queryid。		内部哈希码，从语句的解析树计算得来。
dbname。		数据库名称。
rows。		语句检索或影响的总行数。
calls。		语句被执行的次数。
datid。		这个后端连接到的数据库的 OID。
pid。		这个后端的进程 ID。
leader_pid。		并行组组长的进程 ID，如果该进程是并行查询工作者。如果该进程是一个并行组的组长或不参与并行查询，则为 NULL。
usesysid。		登录到这个后端的用户的 OID。
application_name。		连接到这个后端的应用的名称。
client_addr。		连接到这个后端的客户端的 IP 地址。如果这个字段为空，它表示客户端通过服务器机器上的一个 Unix 套接字连接或者这是一个内部进程，如自动清理。
client_hostname。		已连接的客户端的主机名，由 client_addr 的反向 DNS 查找报告。这个字段将只对 IP 连接非空，并且只有 log_hostname 被启用时才会非空。
client_port。		客户端用于与此后端通信的 TCP 端口号，如果使用 Unix 套接字，则为-1。如果该字段为空，它表示这是一个内部服务器进程。
backend_start。		这个进程被启动的时间。对客户端后端来说，这就是客户端连接到服务器的时间。
xact_start。		这个进程的当前事务被启动的时间，如果没有活动事务则为空。如果当前查询是它的第一个事务，这一列等于 query_start 列。
query_start。		当前活动查询被开始的时间，如果 state 不是 active，则为上一个查询开始的时间。
state_change。		state 上一次被改变的时间。

wait_event_type	后端等待的事件类型，如果有的话;否则 NULL。
wait_event	如果后端当前正在等待，则等待事件名称，否则为 NULL。
state	这个后端的当前总体状态。可能的值为： active: 后端正在执行一个查询。 idle: 后端正在等待一个新的客户端命令。 idle in transaction: 后端在一个事务中，但是当前没有正在执行一个查询。 idle in transaction (aborted): 这个状态与 idle in transaction 相似，除了在该事务中的一个语句导致了一个错误。 fastpath function call: 后端正在执行一个 fast-path 函数。 disabled: 如果在这个后端中 track_activities 被禁用，则报告这个状态。
backend_xid	这个后端的顶层事务标识符，如果存在。
backend_xmin	当前后端的 xmin 范围。
backend_type	当前后端的类型。可能的类型为 autovacuum launcher, autovacuum worker, logical replication launcher, logical replication worker, parallel worker, background writer, client backend, checkpoint, startup, walreceiver, walsender and walwriter. 除此以外，由扩展注册的后台 Worker 可能有额外的类型。
total_exec_time	执行语句所花费的总时间，以毫秒为单位。
query	这个后端最近查询的文本。如果 state 为 active，这个字段显示当前正在执行的查询。在所有其他状态下，它显示上一个被执行的查询。默认情况下，查询文本会被截断至 1024 个字节，这个值可以通过参数 track_activity_query_size 更改。

TOP SQL

TOP SQL 是对最近 24h 查询语句的多维统计排行榜；基于 query md5 来对语句分组，主要排行维度有：总调用次数、总执行耗时、最小执行时间、最大执行时间、平均执行时间、总体时间标准差、总影响行数、总读取时间、总写入时间。

LightDB 云控制台

单实例

集群管控

集群列表

主机管理

实例列表

集群拓扑图

集群拓扑图

主机监控与管理

主机概览

数据库监控与管理

PWR管理

数据库概览

实时SQL

TOP SQL

10.19.69.255:5555

页面更新于: 2022-06-21 10:34:35

自动刷新

时间范围

排序字段

数据库名称

请输入

用户名称

请输入

Query MD5

请输入

查询

重置

请输入列表标题名称

共 0 条

数据库名称	用户名称	查询语句	Query MD5	总调用次数	总执行耗时	最小执行时间	最大执行时间	平均执行时间	总体时间标准差	总影响行数	总读取时间	总写入时间
postgres	lightdb	SELECT coll	a1d10d07c	84186	589130.032	5.364	21909.649	6.998	1300.598	84186	17073.630	0
postgres	lightdb	delete from	885910bd6	8337	465766.018	39.298	130.041	55.867	12.739	86055	97.609	0
postgres	lightdb	select jrd.jo	4d68cc5b0	2603	385223.592	128.131	1135.969	147.992	196.822	15000	1049.312	0
postgres	lightdb	select coun	9fd69f90c1	7705	257200.428	24.103	424.765	33.381	50.868	7705	1022.871	0
postgres	lightdb	SELECT lt_c	a6e5f9cd06	142	143814.650	709.790	14811.083	1012.779	0	142	6124.270	1.111
em	lightdb	insert into l	a14eb5cbd	4911741	141009.974	0.013	7.548	0.029	0.034	4911741	0.625	0
em	lightdb	select cron	4127ce0c9	8382042	74254.245	0.001	1.091	0.009	0.004	3470274	0.024	0
em	lightdb	select host	6043f3fbeb	8382113	62222.979	0.002	1.087	0.007	0.003	3470315	0.050	0
postgres	lightdb	SELECT nan	7db984414	20023	24132.879	1.025	2.353	1.205	0.136	60069	0	0

共 0 条

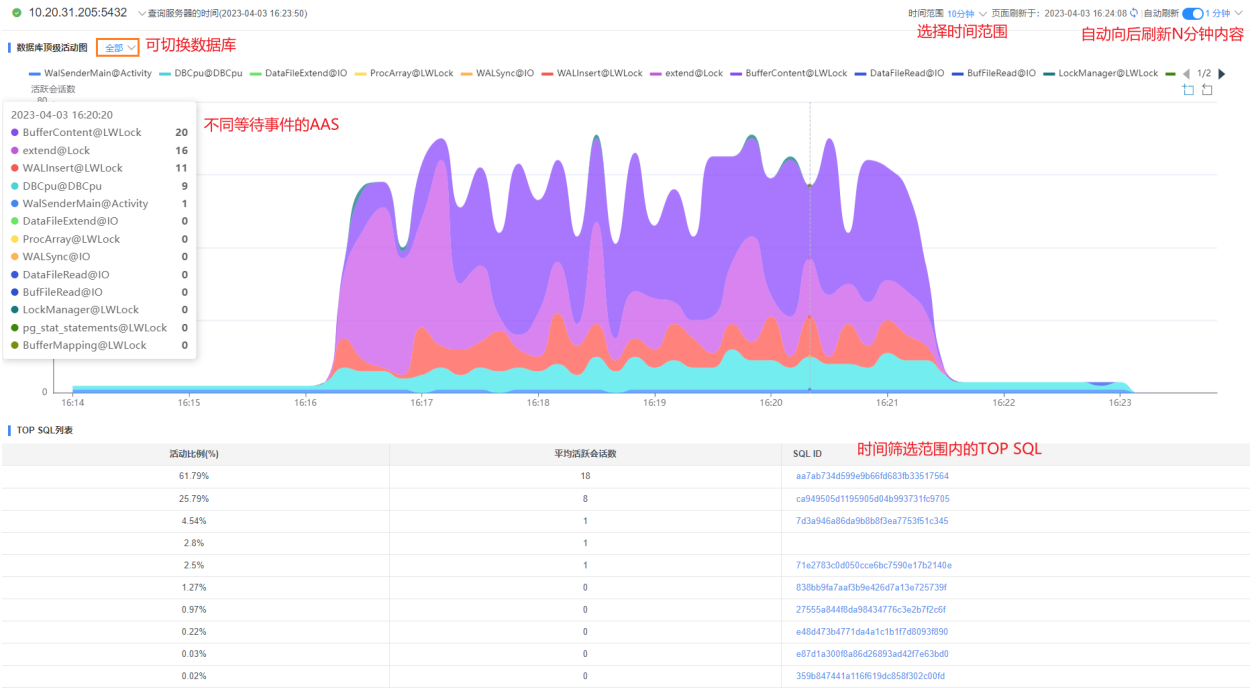
50条/页

跳至

页

顶级活动

顶级活动展示不同等待事件的平均活跃会话情况，横轴为时间，每 10 秒一个间隔，竖轴为活跃会话数。

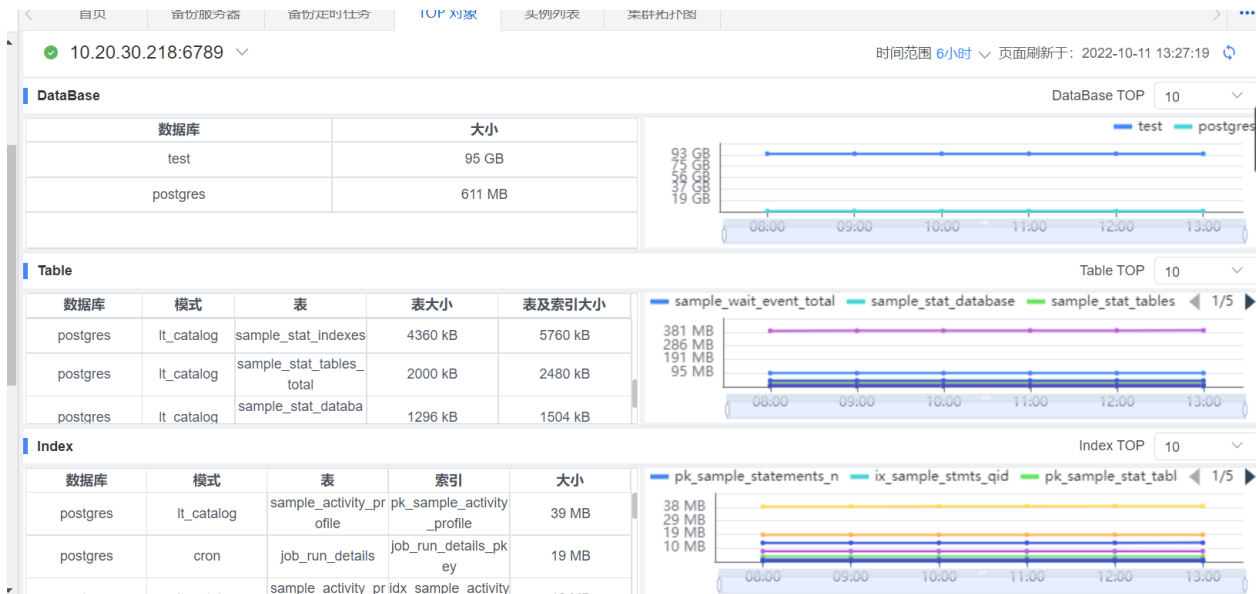


TOP SQL 列表展示一定时间内 SQL 活动占比, 平均活跃会话数, TOP SQL 列表和平均活跃会话筛选的时间范围联动, 点击具体 SQL ID 可以下钻获取具体的 SQL 内容, 同时可在线获取该 SQL 的执行计划。



TOP 对象

TOP 对象是对数据库大小，数据库表大小，索引大小进行排行；



索引监控

合理的索引能大幅提升性能，但冗余的索引也会降低数据库性能，索引监控模块会扫描数据库中存在无效的索引，可对索引组织提供一定的参考。

10.20.30.199:5438 页面刷新于: 2023-04-04 16:39:54 自动刷新

数据库名称 postgres 表名称 请输入 查询

schema名	表名	表行数	表大小	索引名	索引大小	唯一索引	被扫描次数	扫描记录数	回表扫
abc	ttt	0	NaNB		0B				
cron	job	5	NaNB	job_pkey	NaNB		91204	91204	
cron	job	5	NaNB	jobname_username_u	NaNB		0	0	
cron	job_run_details	100008	NaNB	job_run_details_pkey	NaNB		5722862	8693699490	
cron	lt_job_ext	5	NaNB	jobid_username_uniq	NaNB		0	0	
hint_plan	hints	0	NaNB	hints_pkey	NaNB		0	0	
hint_plan	hints	0	NaNB	hints_norm_and_app	NaNB		0	0	
public	emp	0	NaNB		0B				
public	keke	39602280	NaNB		0B				
public	t1	4	NaNB	id	NaNB		0	0	
public	t1	4	NaNB	t1_id_idx	NaNB		0	0	

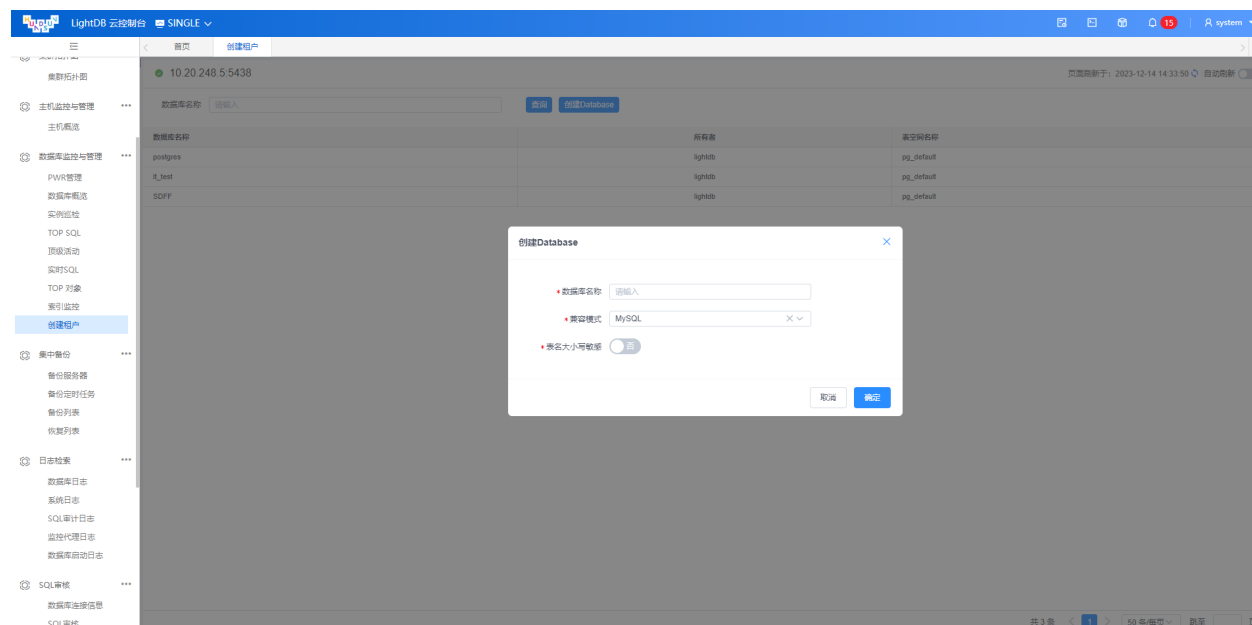
共 27 条 1 50 条/每页 跳至 页

创建租户

创建租户功能是可以通过 em 在线创建 database。

创建租户具有以下限制：

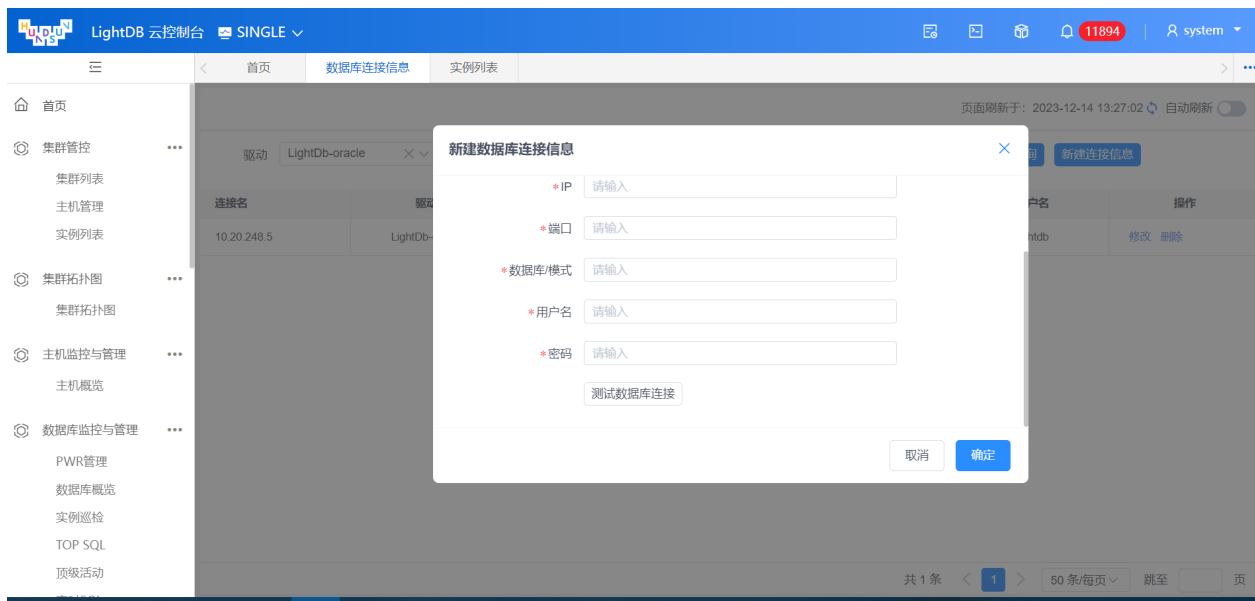
1. 在线创建租户，当数据库版本号小于 22.4，可以填写数据库名称；
2. 数据库版本号大于等于 22.4 小于 23.3，可以填写数据库名称和选择兼容模式；
3. 数据库版本号大于等于 23.3 可以填写数据库名称和选择兼容模式并在选择 mysql 模式后可以选择是否开始表名大小写敏感按钮；
4. 当在分布式的 cn 节点创建数据库，会同步到 dn 节点上，当在 dn 节点在创建数据库，则只在 dn 上创建租户。
5. 数据库实例角色为 standby 和 witness 不支持创建租户。



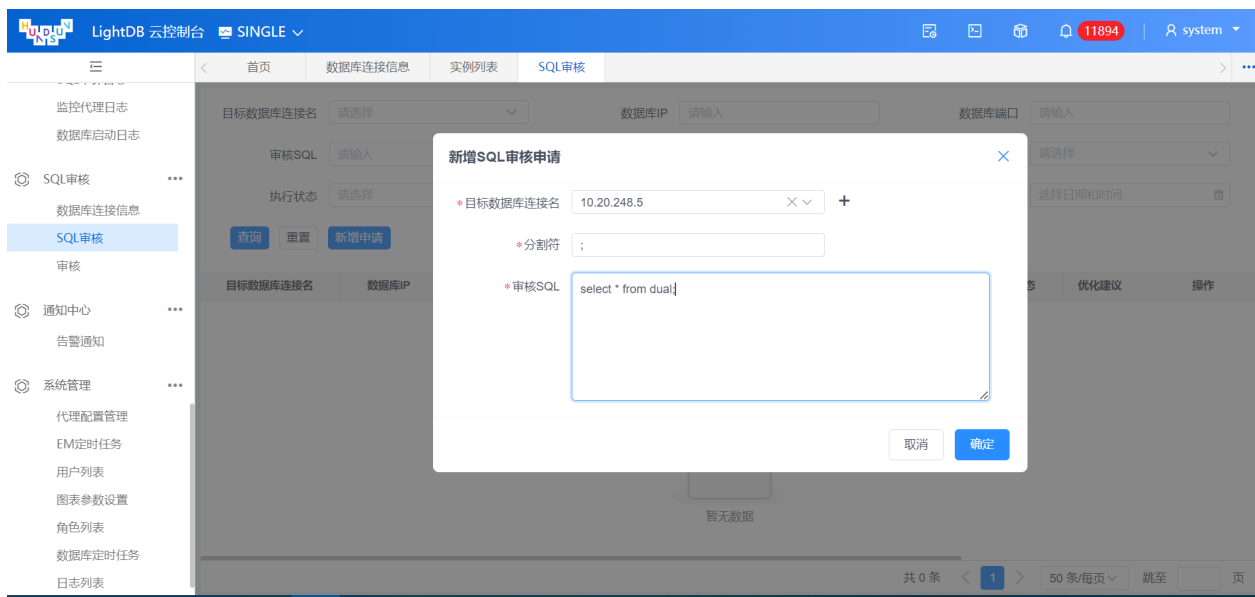
SQL 审核

SQL 审核功能是可以新建数据库连接信息，然后新增 SQL 申请的时候可以指定数据库连接信息提交审核工单，后面可以通过人工来审核 SQL 工单是否通过。

1, 新增数据库连接信息，目前支持 POSTGRESQL、ORACLE、MYSQL(5.7、8.0)、OCEANBASE(mysql、oracle)、TDSQL(5.7、8.0)、DM、OPENGAUSS, lightdb-mysql, lightdb-oracle 这些类型的数据库



2, 新增 SQL 审核申请, 指定数据库连接信息新增 SQL 审核工单

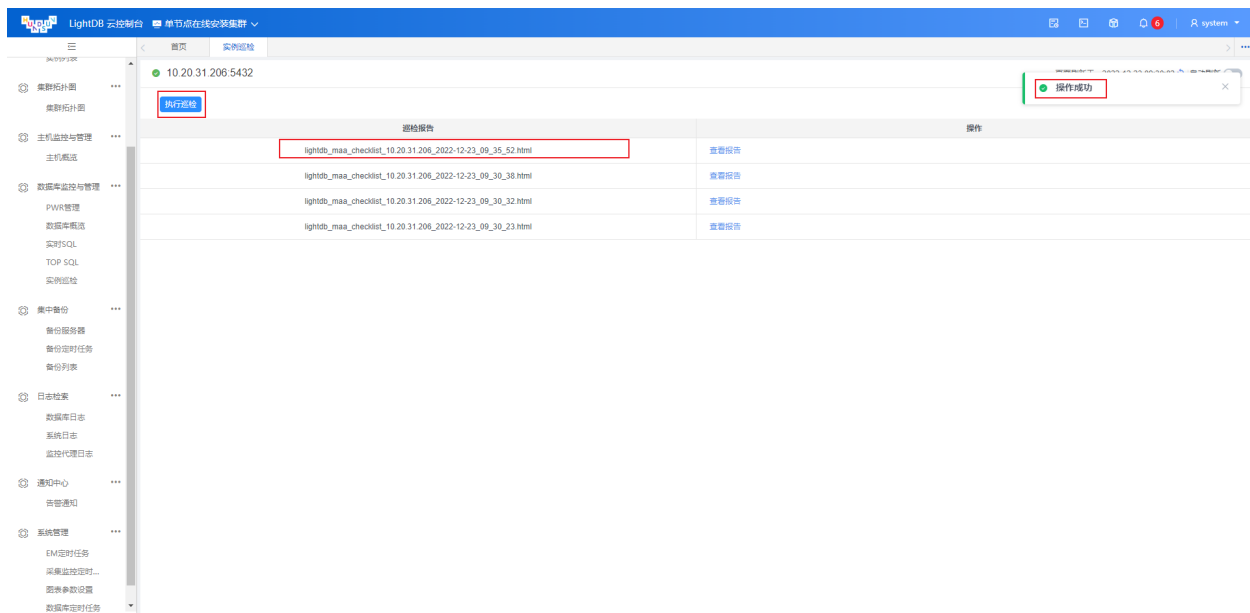


3, 人工审核, 审核新增的 SQL 工单

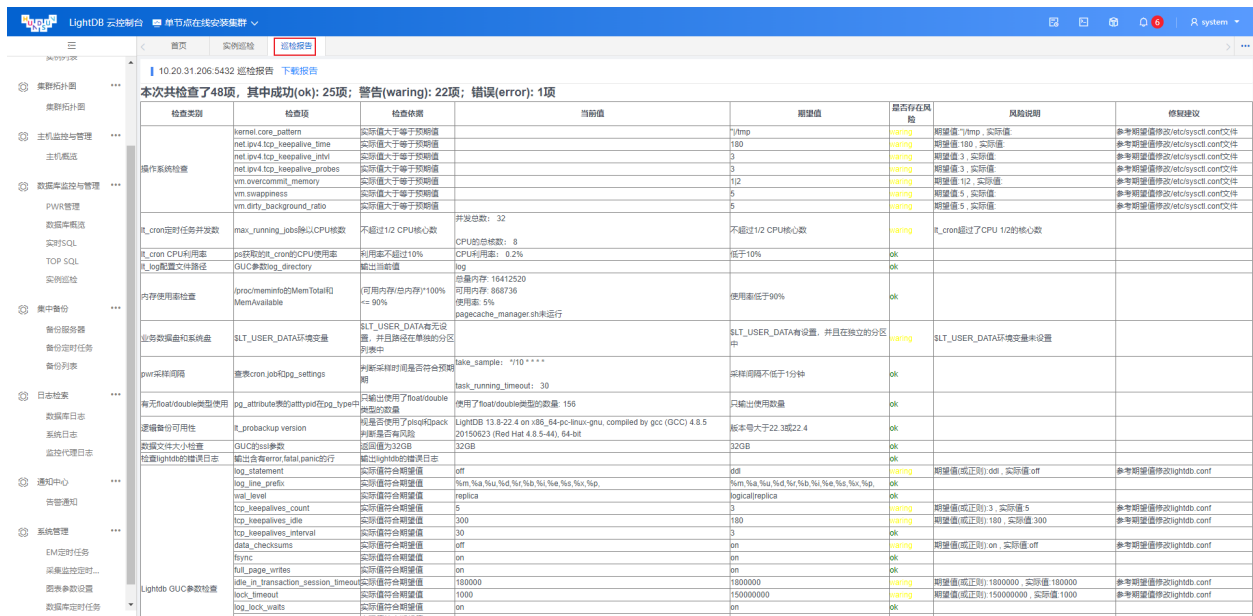


实例巡检

由于巡检工具使用 python3 编写，需要访问 lightdb 查询 GUC 参数以及节点状态，故务必确保系统安装了 python3。注：自 LightDB 22.4 版本开始，LightDB 依赖 python3 版本。1、生成实例的巡检报告



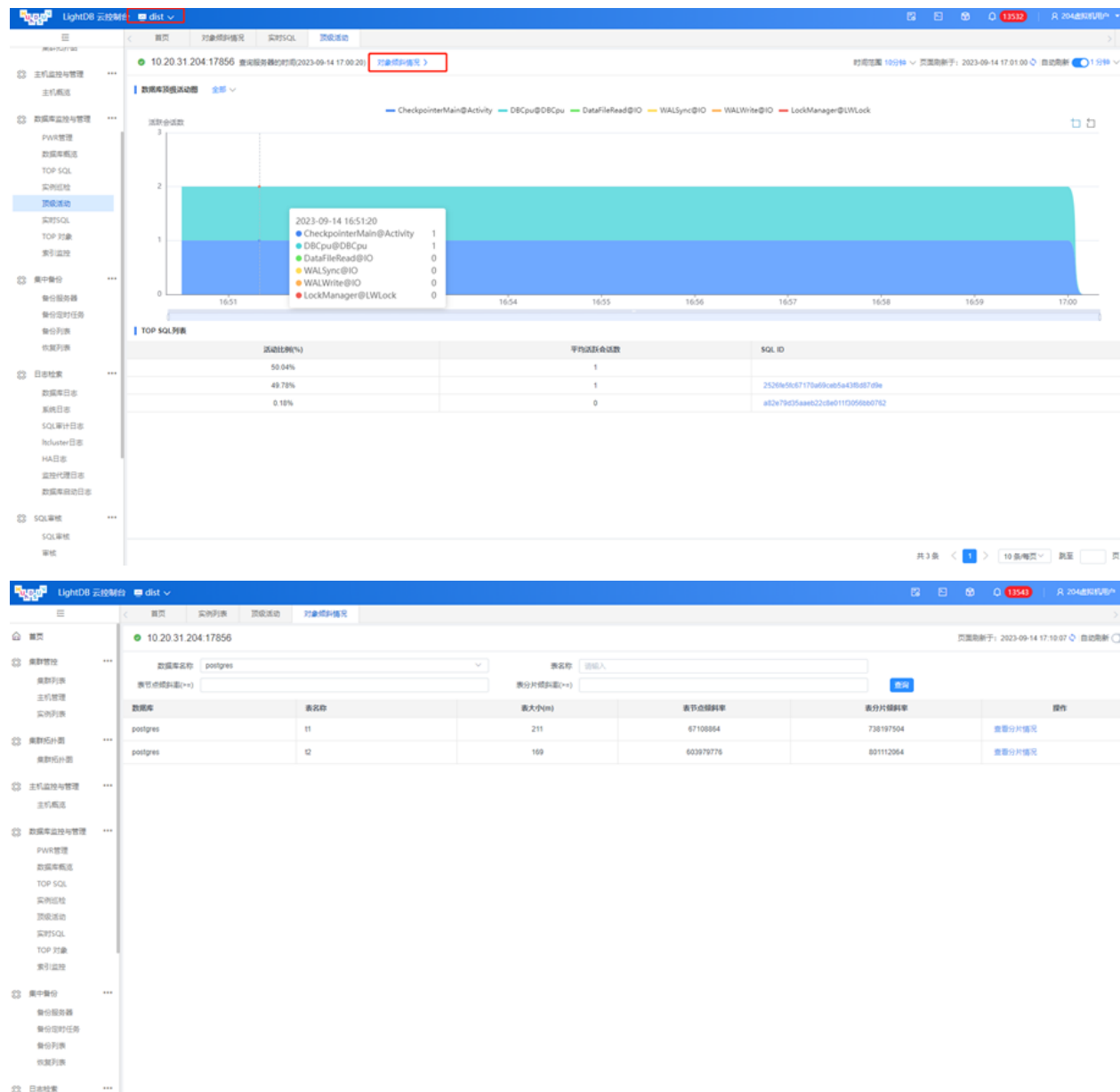
2、查看实例巡检报告内容

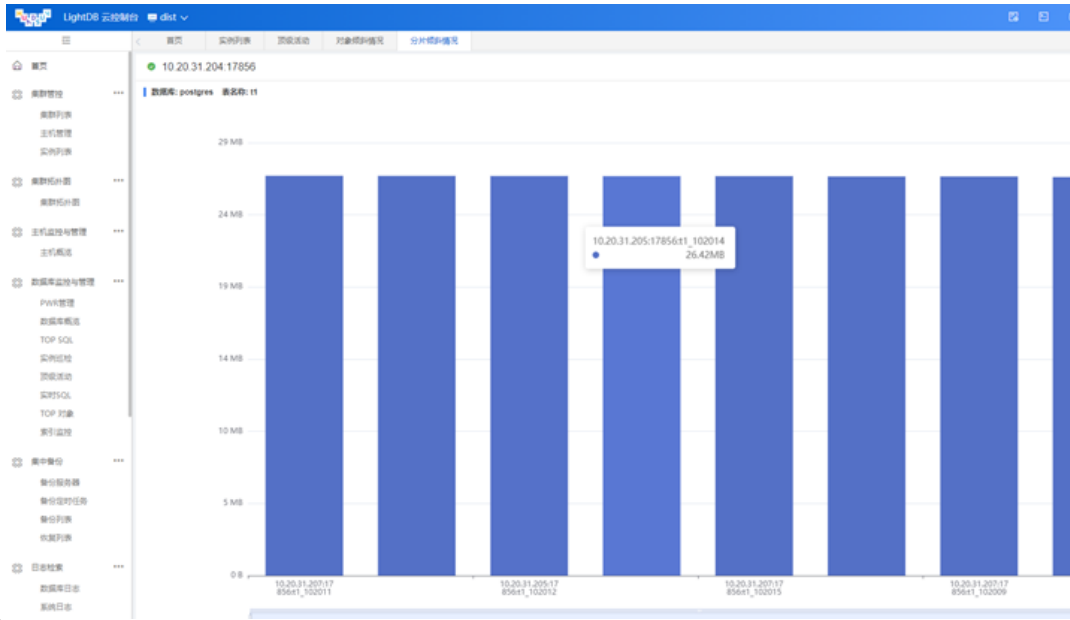


67

对象倾斜率情况

如果代理监控的实例是分布式数据库实例，则可以查看每个数据库的倾斜率信息，以及分片信息。





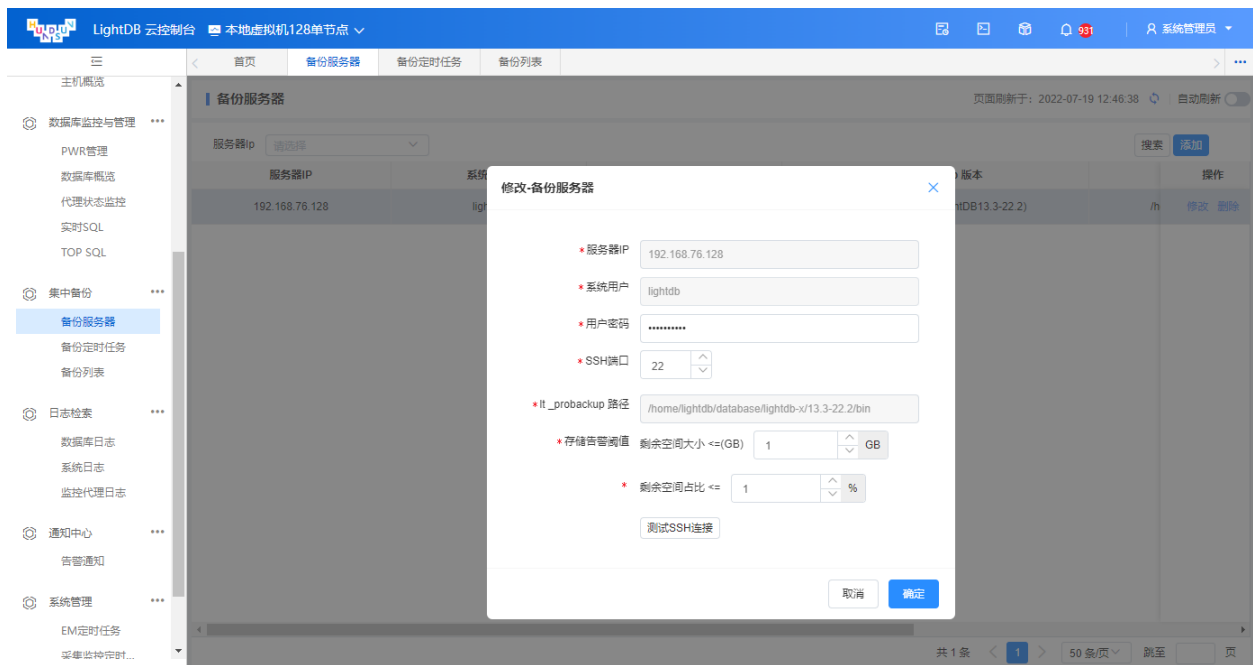
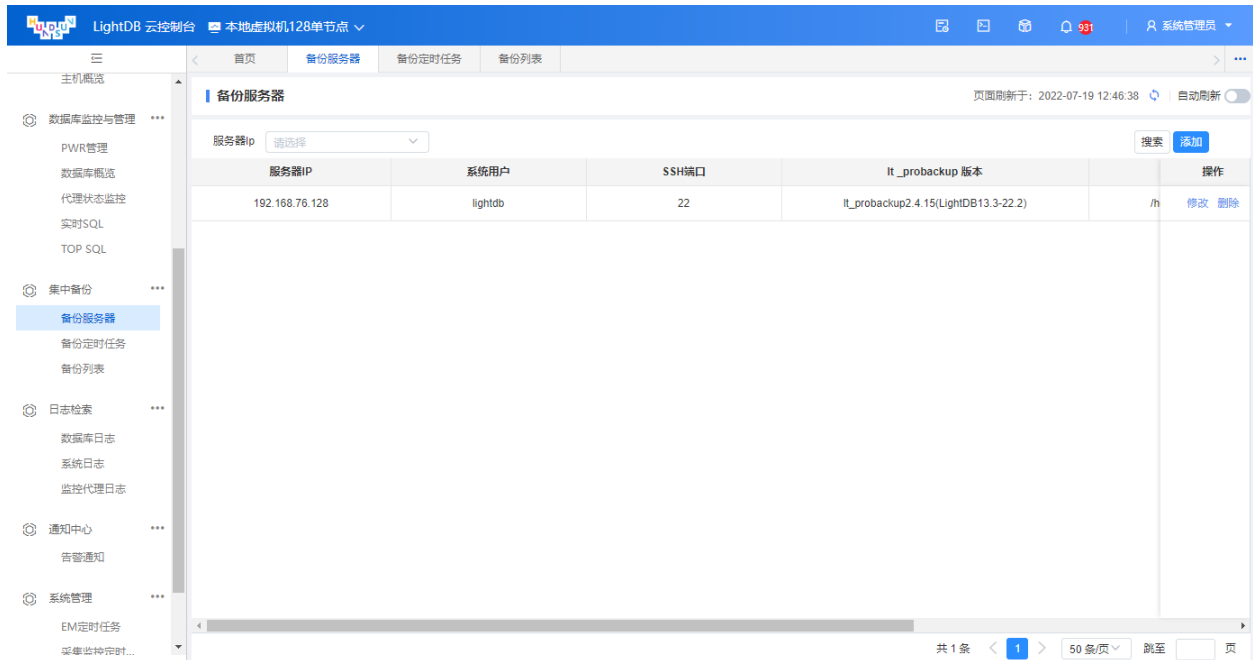
点击查看分片情况查看分片信息

3.10 集中备份

集中备份基于 lt_probackup 实现数据库实例的数据备份和 WAL 归档；一个备份服务器配备一个备份实例，目前仅支持备份服务器的备份实例与被备份实例同版本的情况。一个备份服务器可以初始化多个备份目录，一个备份目录下可以添加多个实例（一个实例只允许被添加一次），一个实例可以执行多次备份，增量备份之前必须有成功的全量备份。

备份服务器

备份服务器是用于存储备份文件的服务器，系统提供自定义存储告警阈值，触发阈值将会发送告警通知。添加备份服务器前准备：1. 参与备份的服务器平台要一致，如 el7.x87 或者 KylinV10sp1.arm。2. 预先在备份服务器安装 lightdb，版本与备份实例需一致。



注意：【It_probackup 路径】表示 It_probackup 这个命令在备份服务器上的绝对路径 (该路径为目录不包含 It_probackup)

备份定时任务

备份定时任务开放定时上线，仅支持手动执行备份（增量备份前需要已成功全量备份），支持手动删除备份。添加备份任务前准备：1. 已部署备份实例。2. 在备份实例配置数据库用户 `lt_probackup` 的访问权限和免密，编辑 `$LTDATA/lt_hba.conf` 添加如下内容：host all lt_probackup 10.20.128.128/32 trust 字段内容说明：host：此记录匹配使用 TCP/IP 进行的连接尝试。all：匹配所有的数据库 lt_probackup：特定的数据库用户名称（该用户用于备份）10.20.128.128/32：客户端计算机地址。该地址下的客户端允许访问数据库 trust：无条件允许连接。即允许客户端地址为 10.20.128.128/32 的数据库用户 lt_probackbap 访问（不需要密码）。

```
# "local" is for Unix domain socket connections only
local    all                all                                trust
# IPv4 local connections:
host     all                all                                127.0.0.1/32      trust
host     all                lt_probackup                    192.168.76.128/32 trust
host     all                all                                0.0.0.0/0        mds
# IPv6 local connections:
host     all                all                                ::1/128          trust
# Allow replication connections from localhost, by a user with the
# replication privilege.
local    replication        all                                trust
host     replication        all                                127.0.0.1/32     trust
host     replication        all                                ::1/128          trust
```

提示：未在数据库实例的 `$LTDATA/lt_hba.conf` 文件中配置数据库用户 `lt_probackup` 访问免密，则为该数据库实例添加备份定时任务失败，失败提示信息为“需要在 `$LTDATA/lt_hba.conf` 配置备份用户 `lt_probackup` 的访问权限和免密”。22.1 版本的数据库实例，为该实例添加备份定时任务失败，错误提示信息中 `$LTDATA` 对应 22.1 版本的数据库实例为 `$PGDATA`。lt_hba.conf 对应 22.1 版本的数据库实例为 `pg_hba.conf`。3.em 为备份实例配置持续 WAL 归档（备份基于持续 WAL 归档）。需要重启数据库实例服务保证配置的持续 WAL 归档生效。4. 使用命令行 `ltsql` 连接到数据库实例，查看配置项（`archive_mode`、`wal_level`、`archive_command`、`archive_timeout`）内容：

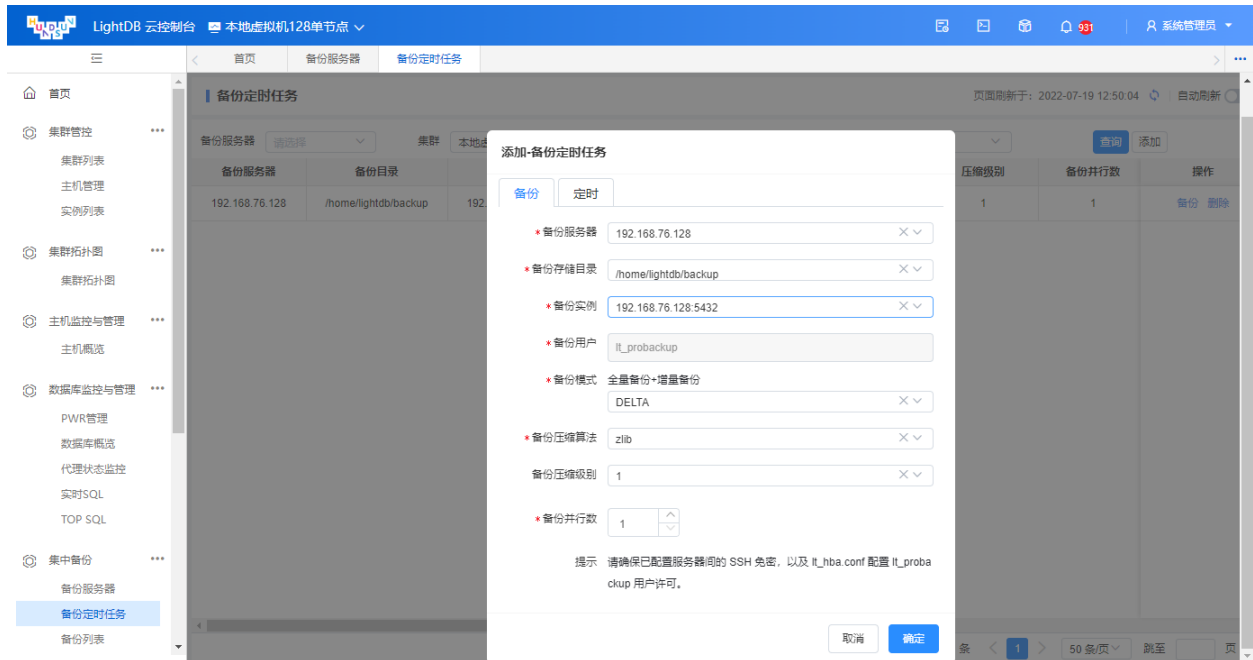
```
show archive_mode;
show wal_level;
show archive_timeout;
show archive_command;
```

5. 配置备份服务器主机与备份实例主机之间 SSH 双向免密互通；

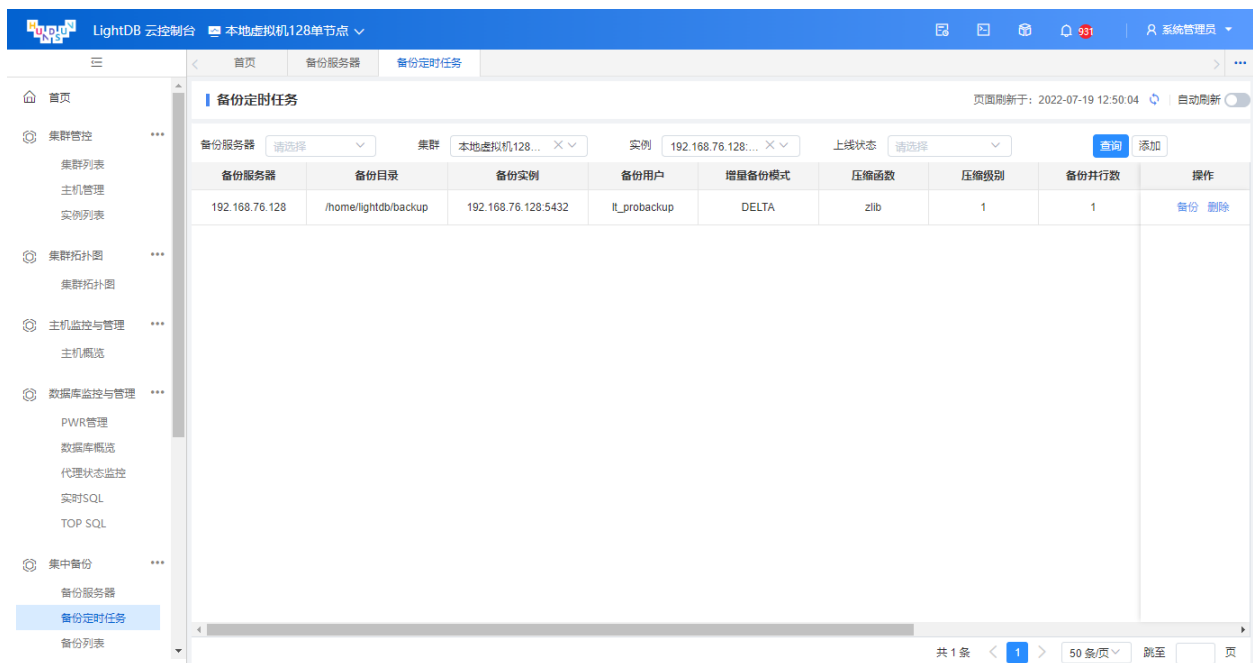
```
ssh-keygen -t rsa -P "" -f ~/.ssh/id_rsa
ssh-copy-id lightdb@backup_server_ip

ssh-keygen -t rsa -P "" -f ~/.ssh/id_rsa
ssh-copy-id lightdb@instance_server_ip
```

6. 远程备份的并行数受 SSH 连接限制，因为 SSH 连接数默认上限为 10，当备份并行数超过 10 时，需要修改被备份实例所在主机的 SSH 连接数限制，参考如下（超级用户执行）：`vim /etc/ssh/sshd_config` 修改 `MaxStartup`、`MaxSessions` 参数，将最小连接数设置为 80（建议值为最大并行数 +10）。保存后退出。重启 `sshd` 服务：`systemctl restart sshd` 或者 `service sshd reload`。MaxSessions 80 MaxStartup 80



备份模式 (增量) 说明: DELTA: 在这种模式下, `lt_probackup` 命令读取数据库实例的 DATA 目录中的所有数据文件, 并仅复制自上次备份以来发生更改的那些页面。PAGE: `lt_probackup` 命令会从前一次完整备份或增量备份开始扫描归档文件中的所有 WAL 文件。新创建的备份只包含 WAL 记录中提到的页面。这要求自上一次备份以来的所有 WAL 文件都存在于 WAL 存档中。要求配置持续 WAL 归档。



手工执行备份接入日志与进度

LightDB 云控制台

节点所在安装集群

首页

实例监控

性能报告

备份服务器

备份定时任务

备份列表

三

主机管理

实例列表

集群拓扑图

集群后补图

主机监控与管理

主机概览

数据库监控与管理

PVW管理

数据库概览

实时SQL

TOP SQL

实例巡检

集中备份

备份服务器

备份定时任务

备份列表

恢复列表

日志检索

数据库日志

系统日志

监控代理日志

通知中心

消息通知

系统管理

EM定时任务

备份定时任务

页面更新于: 2022-12-23 09:57:12 自动刷新

备份服务器

备份目录

备份实例

备份用户

增量备份模式

压缩函数

压缩级别

备份并行数

备份窗口(天数)

全量备份留存副本数

备份每天定时

上线状态

操作

10.20.31.206	/usr/local/lightdb/back	10.20.31.206.5432	il_probackup	DELTA	zlib	1	2	14	1	0 00 01 ***	上线	下线	手工	删除
--------------	-------------------------	-------------------	--------------	-------	------	---	---	----	---	-------------	----	----	----	----

共 1 条 < 1 > 50 条/页 跳至 页

LightDB 云控制台

节点所在安装集群

首页

实例监控

性能报告

备份服务器

备份定时任务

备份列表

三

主机管理

实例列表

集群拓扑图

集群后补图

主机监控与管理

主机概览

数据库监控与管理

PVW管理

数据库概览

实时SQL

TOP SQL

实例巡检

集中备份

备份服务器

备份定时任务

备份列表

恢复列表

日志检索

数据库日志

系统日志

监控代理日志

通知中心

消息通知

系统管理

EM定时任务

备份定时任务

页面更新于: 2022-12-23 09:57:12 自动刷新

备份服务器

备份目录

备份实例

备份用户

增量备份模式

压缩函数

压缩级别

备份并行数

备份窗口(天数)

全量备份留存副本数

备份每天定时

上线状态

操作

10.20.31.206	/usr/local/lightdb/back	10.20.31.206.5432	il_probackup	DELTA	zlib	1	2	14	1	0 00 01 ***	上线	下线	手工	删除
--------------	-------------------------	-------------------	--------------	-------	------	---	---	----	---	-------------	----	----	----	----

共 1 条 < 1 > 50 条/页 跳至 页

手动执行备份

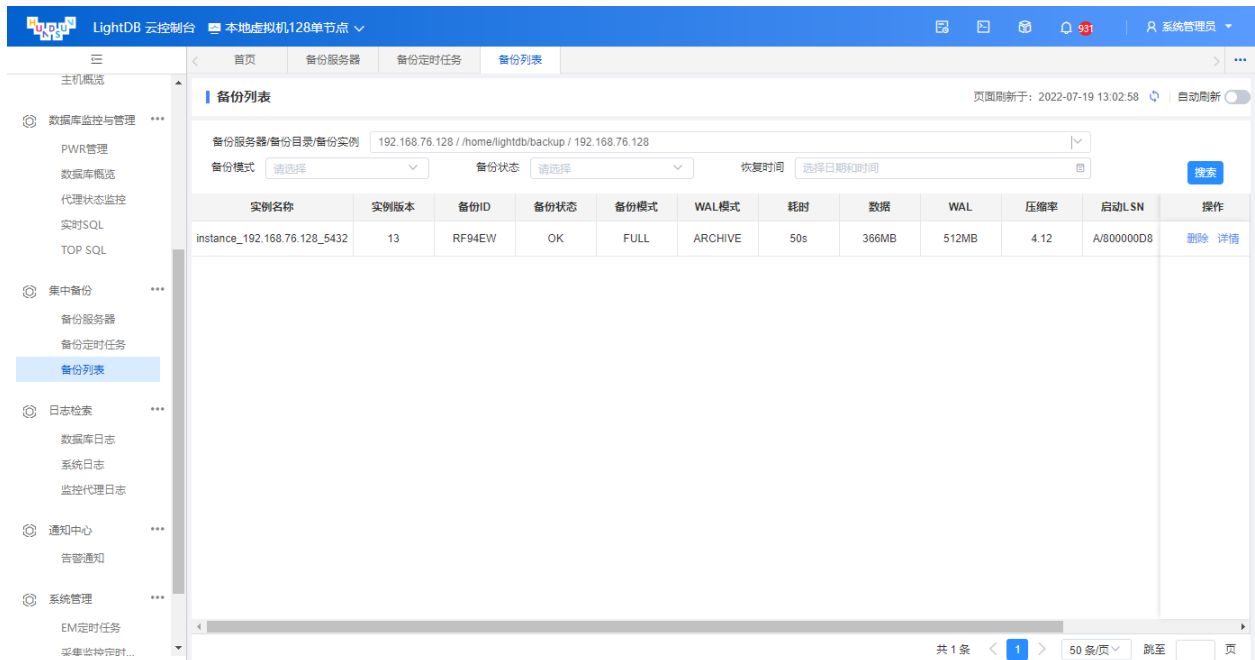
本次备份模式 全量

取消 确定



备份列表

备份列表页面展示备份服务器-备份目录下的备份列表，点击“详情”可以查看备份详情。



备份状态说明：OK：备份完成并且有效 DONE: 备份已完成，但尚未验证。RUNNING: 备份正在进行中。MERGING: 备份正在合并。MERGED: 备份数据文件已成功合并，但其元数据正在更新过程中。只有全量备份才能具有此状态。CORRUPT：一些备份文件已经破坏。DELETING：正在删除备份文件。ERROR: 备份由于意外错误而中止。ORPHAN: 备份无效，因为其父备份之一已损坏或丢失。

LightDB 云控制台 本地虚拟机128单节点

备份列表

备份服务器/备份目录/备份实例: 192.168.76.128 / /home/lightdb/backup / 192.168.76.128

备份模式: 选择

备份详情: instance_192.168.76.128_5432-RF94EW

属性	值
data-bytes	383330659
start-time	2022-07-19 12:56:56+08
compress-level	1
compress-alg	zlib
end-time	2022-07-19 12:57:46+08
timelineid	1
wal-bytes	536870912
recovery-xid	2337164
program-version	2.4.15
uncompressed-bytes	1581094711
recovery-time	2022-07-19 12:57:40+08
block-size	8192
primary_conninfo	user=lt_probackup channel_binding=prefer host=192.168.76.128 port=5432 sslmode=prefer sslcompression=0 ssl_min_protocol_version=TLSv1.2 gssencmode=disable krb_srvname=postgres target_session_attrs=any
backup-mode	FULL

WAL: 512MB 压缩率: 4.12 启动LSN: A/800000D8 操作: 删除 详情

备份删除

备份删除策略：仅合并过期备份。删除 wal 文档(可选) 仅删除过期备份。删除 wal 文档(可选) 指定备份状态删除。删除 wal 文档(可选) CORRUPT: 一些备份文件已经破坏。ERROR: 备份由于意外错误而中止。ORPHAN: 备份无效，因为其父备份之一已损坏或丢失。指定备份 id 记录删除。删除 wal 文档(可选)

LightDB 云控制台 单节点在线部署集群

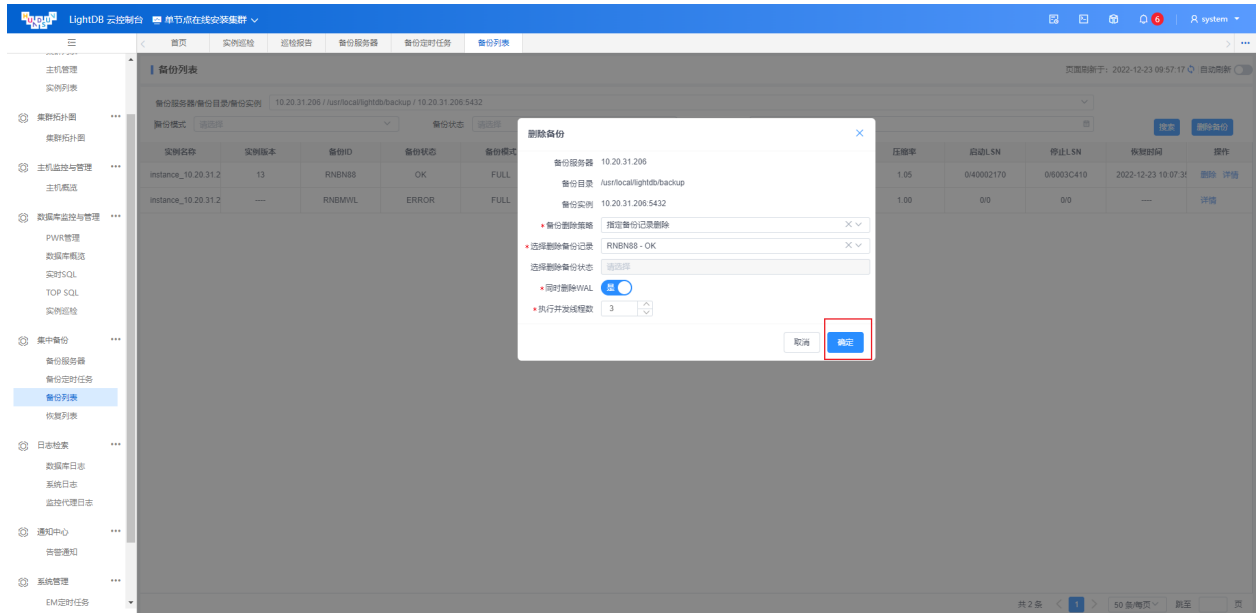
备份列表

备份服务器/备份目录/备份实例: 10.20.31.206 / /usr/local/lightdb/backup / 10.20.31.206 5432

备份模式: 选择 备份状态: 选择 恢复时间: 选择日期和时间

实例名称	实例版本	备份ID	备份状态	备份模式	WAL模式	耗时	数据	WAL	压缩率	启动LSN	停止LSN	恢复时间	操作
instance_10.20.31.2	13	RNBAB8	OK	FULL	ARCHIVE	21s	819MB	512MB	1.95	0/40002170	0/80003410	2022-12-23 10:07:31	删除 详情
instance_10.20.31.2	---	RNBAMVL	ERROR	FULL	ARCHIVE	10m.1s	0	0	1.00	0/0	0/0	---	详情

共2条 < 1 > 50条/页 跳至 页



恢复列表

恢复前准备：恢复实例所在服务器、备份服务器，需要满足以下条件：1. 主机之间配置 SSH 双向免密互通；

```
ssh-keygen -t rsa -P "" -f ~/.ssh/id_rsa
ssh-copy-id lightdb@backup_server_ip

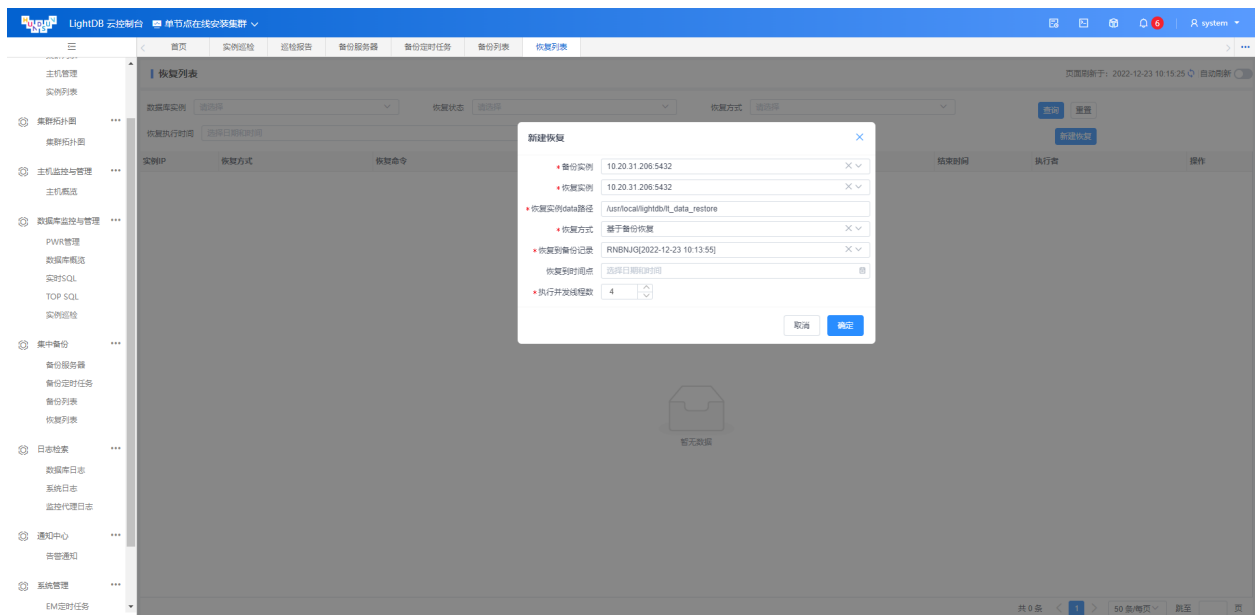
ssh-keygen -t rsa -P "" -f ~/.ssh/id_rsa
ssh-copy-id lightdb@restore_instance_server_ip
```

2. 远程还原的并行线程数受 SSH 连接限制，因为 SSH 连接数默认上限为 10，当并行数超过 10 时，需要修改被恢复实例所在主机的 SSH 连接数限制，参考如下（超级用户执行）：

```
vim /etc/ssh/sshd_config
#修改 MaxStartup、MaxSessions参数，将最小连接数设置为 80
→80（建议值为最大并行数+10）。保存后退出。
MaxSessions 80
MaxStartup 80
```

重启 sshd 服务：systemctl restart sshd 或者 service sshd reload。

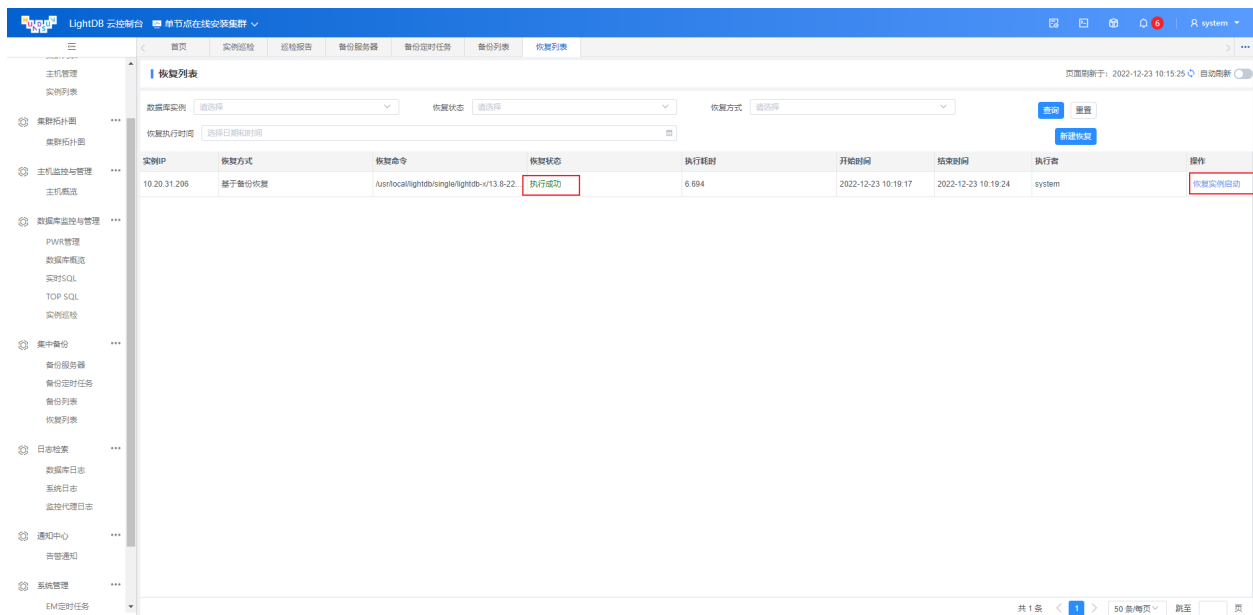
执行恢复：恢复方式有两种：基于时间点恢复、基于备份恢复，选择基于时间点，则需要指定恢复的时间点，选择基于备份，则需要指定恢复的备份 ID；基于备份 ID 的恢复方式如下：



新建恢复已接入日志与进度

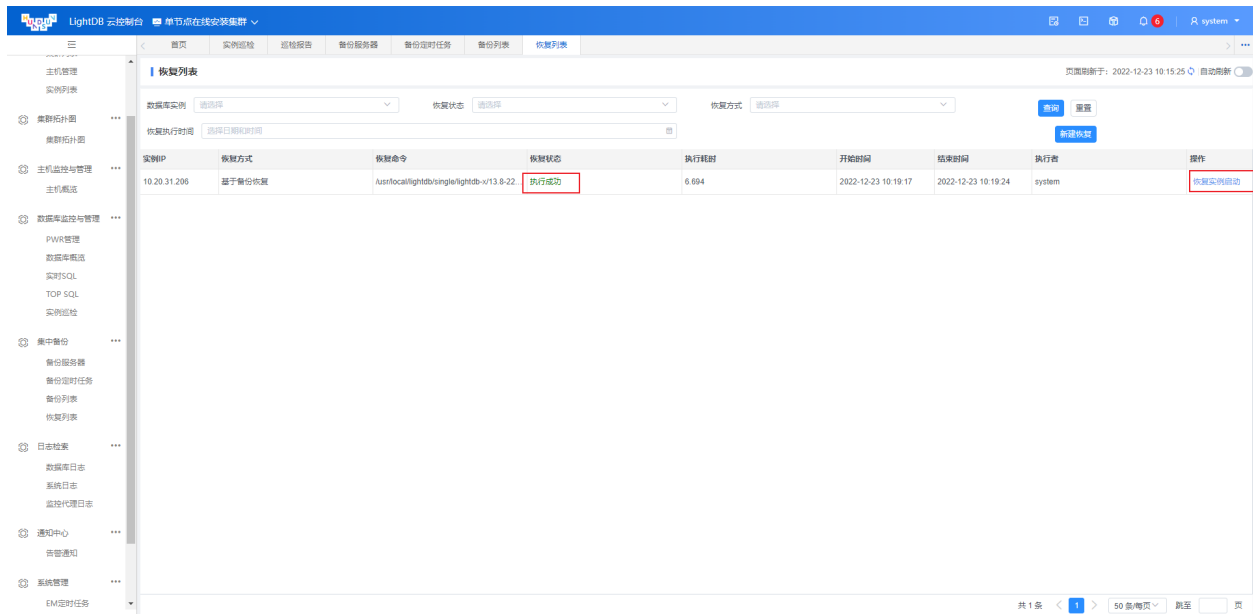


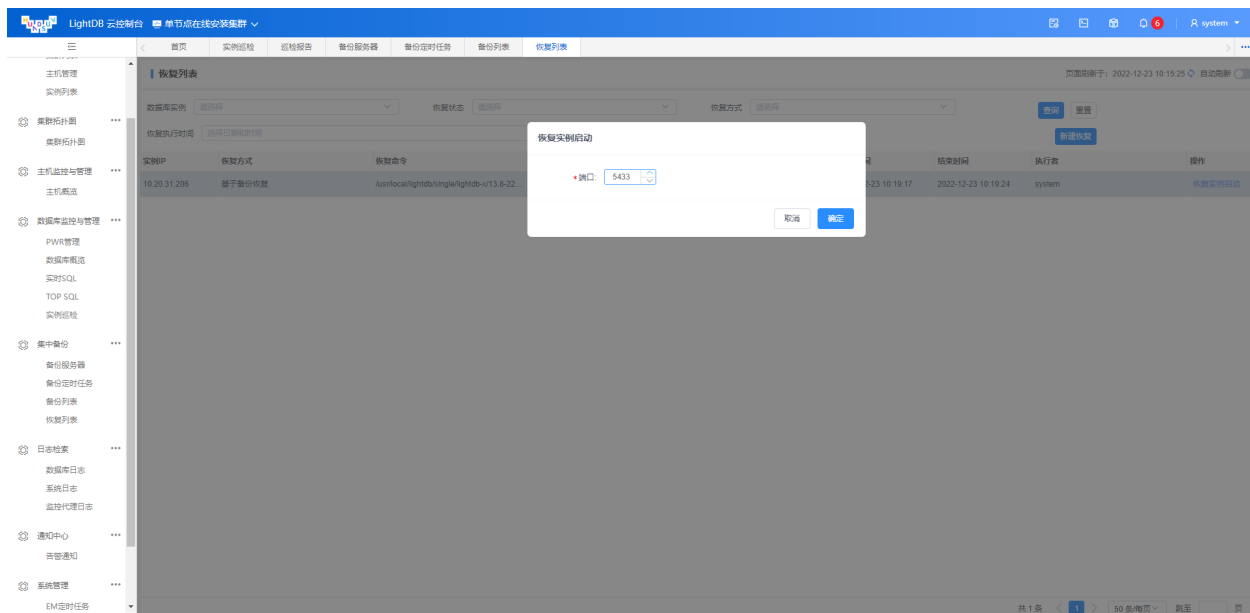
在当前页面可以查询本次执行恢复的状态：



恢复后立即启动

可以针对执行恢复成功的情况，执行恢复后立即启动，基于恢复后实例目录启动一个单实例，该实例可以后续再添加到 em 监控





3.11 日志检索

数据库日志

数据库日志定时从 LightDB 数据库日志文件中采集。日志文件地址 \$LTDATA/log/。

```
[lightdb@localhost data]$  
[lightdb@localhost data]$ cd $PGDATA/log  
[lightdb@localhost log]$ ls -l  
总用量 254644  
-rw----- 1 lightdb lightdb 208329 1月 25 14:36 lightdb-2022-01-25_143320.log  
-rw----- 1 lightdb lightdb 7770 1月 25 14:37 lightdb-2022-01-25_143652.log  
-rw----- 1 lightdb lightdb 6069161 1月 26 00:00 lightdb-2022-01-25_143826.log  
-rw----- 1 lightdb lightdb 13121824 1月 27 00:00 lightdb-2022-01-26_000000.log  
-rw----- 1 lightdb lightdb 71435518 1月 28 00:00 lightdb-2022-01-27_000000.log  
-rw----- 1 lightdb lightdb 127025056 1月 29 00:00 lightdb-2022-01-28_000000.log  
-rw----- 1 lightdb lightdb 41330823 1月 29 08:07 lightdb-2022-01-29_000000.log  
[lightdb@localhost log]$  
[lightdb@localhost log]$
```

数据库日志采集以行为单位进行读取，以段为单位进行采集。每段的规定是以日期开头到下一个日期开头之前的所有行的内容为一。

10.188.131.245:5432		上次请求响应时间:7ms		页面刷新于: 2022-06-29 10:42:29		自动刷新
日志内容	搜索内容	?	日志时间范围	选择日期和时间	搜索	
总览	其他	自动检查点	归档			
过滤条件	ID	时间戳	日志类型	日志级别	日志消息	解析计划
过滤字段	480	2022-06-28 11:15:44	归档	log	archiver process (PID 52273) exited with exit code 1	
日志级别	479	2022-06-28 11:15:44	归档	fatal	archive command failed with exit code 127 The failed archive command was: rm -f /usr/local/lightdb/lightdb-x/13.3-22.2/archive/000000010000000000000001 && ...	
	488	2022-06-28 11:16:14	归档	log	archiver process (PID 52730) exited with exit code 1	
日志类型	487	2022-06-28 11:16:14	归档	fatal	archive command failed with exit code 127 The failed archive command was: rm -f /usr/local/lightdb/lightdb-x/13.3-22.2/archive/000000010000000000000001 && ...	
数据库节点					archiver process (PID 52143) exited with exit code 1	
应用名	478	2022-06-28 11:15:44	归档	log	sh: /usr/local/lightdb/lightdb-x/13.3-22.2/bin/lt_probackup_for_em.sh: No such file or directory	
数据库名	477	2022-06-28 11:15:44	归档	fatal	archive command failed with exit code 127 The failed archive command was: rm -f /usr/local/lightdb/lightdb-x/13.3-22.2/archive/000000010000000000000001 && ...	
后端类型	446	2022-06-28 11:13:54	归档	log	archiver process (PID 50536) exited with exit code 1	
	445	2022-06-28 11:13:54	归档	fatal	archive command failed with exit code 127 The failed archive command was: rm -f /usr/local/lightdb/lightdb-x/13.3-22.2/archive/000000010000000000000001 && ...	
	442	2022-06-28 11:13:40	归档	log	archiver process (PID 50275) exited with exit code 1	
	441	2022-06-28 11:13:40	归档	fatal	archive command failed with exit code 127 The failed archive command was: rm -f /usr/local/lightdb/lightdb-x/13.3-22.2/archive/000000010000000000000001 && ...	
	472	2022-06-28 11:15:12	归档	log	archiver process (PID 51509) exited with exit code 1	
	471	2022-06-28 11:15:12	归档	fatal	archive command failed with exit code 127 sh: /usr/local/lightdb/lightdb-x/13.3-22.2/bin/lt_probackup_for_em.sh: No such file or directory The failed archive command was: rm -f /usr/local/lightdb/lightdb-x/13.3-22.2/archive/000000010000000000000001 && ...	
	464	2022-06-28 11:14:42	归档	log	archiver process (PID 51217) exited with exit code 1	
	463	2022-06-28 11:14:42	归档	fatal	sh: /usr/local/lightdb/lightdb-x/13.3-22.2/bin/lt_probackup_for_em.sh: No such file or directory archive command failed with exit code 127 The failed archive command was: rm -f /usr/local/lightdb/lightdb-x/13.3-22.2/archive/000000010000000000000001 && ...	
	397	2022-06-28 11:10:02	归档	log	archiver process (PID 44859) exited with exit code 1	
	396	2022-06-28 11:10:02	归档	fatal	sh: /usr/local/lightdb/lightdb-x/13.3-22.2/bin/lt_probackup_for_em.sh: No such file or directory archive command failed with exit code 127 The failed archive command was: rm -f /usr/local/lightdb/lightdb-x/13.3-22.2/archive/000000010000000000000001 && ...	
	482	2022-06-28 11:15:50	归档	log	archive command failed with exit code 127	
	481	2022-06-28 11:15:50	归档	fatal	archiver process (PID 52385) exited with exit code 1 The failed archive command was: rm -f /usr/local/lightdb/lightdb-x/13.3-22.2/archive/000000010000000000000001 && ...	
	456	2022-06-28 11:14:10	归档	log	archive command failed with exit code 127	
	455	2022-06-28 11:14:10	归档	fatal	archiver process (PID 50562) exited with exit code 1 The failed archive command was: rm -f /usr/local/lightdb/lightdb-x/13.3-22.2/archive/000000010000000000000001 && ...	
	385	2022-06-28 11:09:00	归档	log	archive command failed with exit code 127	
	384	2022-06-28 11:09:00	归档	fatal	archiver process (PID 42820) exited with exit code 1 The failed archive command was: rm -f /usr/local/lightdb/lightdb-x/13.3-22.2/archive/000000010000000000000001 && ...	
	486	2022-06-28 11:16:14	归档	log	archive command failed with exit code 127 archiver process (PID 52430) exited with exit code 1 sh: /usr/local/lightdb/lightdb-x/13.3-22.2/bin/lt_probackup_for_em.sh: No such file or directory	
共 391 条						50 条/页 跳至 页

日志搜索查询过滤条件如下：日志搜索：根据关键词进行搜索，关键词实现了全文检索；

192.168.86.10:5432

上次请求响应时间:23ms 页面刷新于: 2022-04-04 19:00:00 自动刷新

日志内容terminating

日志时间范围选择日期和时间

搜索

总览

日志全文检索

多关键词&连接实现AND条件检索，例如Light & DB
多关键词|连接实现OR条件检索，例如Light | DB

过滤

输入过

日志级别

日志类型

数据库节点

应用名

数据库名

后端类型

日志内容

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-transaction timeout

terminating connection due to idle-in-

日志信息详情：双击单条日志可查看日志消息详情

10.188.132.12:5432

上次请

日志内容

complete

?

日志时间范围

选择日期和时间

总览

其他

过滤条件

过滤字段

日志级别

日志类型

数据库节点

应用名

数据库名

后端类型

	ID	时间戳	日志类型	日志级别	日志消息
	2776	2022-06-28 14:00:31	其他	log	restartpoint complete : wrote 372 buffers (0.1%); 0 WA
	2386	2022-06-28 13:30:32	其他	log	restartpoint complete : wrote 373 buffers (0.1%); 0 WA
	1982	2022-06-28 13:00:20	其他	log	restartpoint complete : wrote 318 buffers (0.1%); 0 WA

日志详情

×

日志信息

日志级别: log

日志类型: 其他

时间: 2022-06-28 13:30:32

日志信息:

restartpoint **complete**: wrote 373 buffers (0.1%); 0 WAL file(s) added, 0 removed, 0 recycled; write=38.022 s, sync=0.001 s, total=38.041 s; sync files=95, longest=0.001 s, average=0.001 s; distance=6506 kB, estimate=6812 kB

原始日志信息

```
{
  "id": 2386,
  "dbLogType": 0,
  "instanceIp": "10.188.132.12",
  "instancePort": 5432,
  "instancePortIncr": null,
  "instanceName": null,
  "logFile": null,
  "logFilePointer": null,
  "dbLogTime": "2022-06-28 13:44:32",
  "dbLogMessage": "restartpoint complete: wrote 373 buffers",
  "dbLogLevel": 1,
  "dbLogLevelIncr": null,
  "dbLogSql": null,
  "dbLogContent": "",
  "sampleTime": "2022-06-28 13:30:32",
  "timeline": null,
  "timelineIncr": null,
  "logTsv": null,
  "logPrefix": null,
  "applicationName": null,
  "userName": null,
  "databaseName": null,
  "remoteHost": null,
  "backendType": null,
  "commandTag": null,
  "sqlState": null,
  "sessionStart": null,
  "transactionId": null,
  "processId": null
}
```

慢 sql 查询计划：可通过点击慢日志在表格最右一栏的解析按钮查询该条慢日志的查询计划

10.20.31.206:5432

上次请求响应时间:12ms 页面刷新于: 2022-06-30 14:23:01 自动刷新

日志内容

搜索内容

日志时间范围

选择日期和时间

搜索

总览 其他 慢sql

过滤条件	ID	时间戳	日志类型	日志级别	日志消息	解析计划
过滤字段	5731	2022-06-30 13:47:35	慢sql	log	duration: 0.051 ms plan: { "Query Text": "update cron.job_run_details set status = \$1, return_message = \$2, end_time = ...	解析
	5759	2022-06-30 13:48:03	其他	context	SQL statement "update cron.job_run_details set status = \$1, return_message = \$2, end_time = \$3 where runid = \$4"	解析
日志级别	5730	2022-06-30 13:47:35	慢sql	log	duration: 0.028 ms plan: { "Query Text": "update cron.job_run_details set status = \$1, start_time = \$2 where runid = \$3"...	解析
日志类型	5758	2022-06-30 13:48:03	其他	context	SQL statement "update cron.job_run_details set status = \$1, start_time = \$2 where runid = \$3"	解析
数据库节点	5729	2022-06-30 13:47:35	慢sql	log	duration: 0.052 ms plan: { "Query Text": "update cron.job_run_details set job_pid = \$1, status = \$2 where runid = \$3", "...	解析
	5757	2022-06-30 13:48:03	其他	context	SQL statement "update cron.job_run_details set job_pid = \$1, status = \$2 where runid = \$3"	解析
应用名	5728	2022-06-30 13:47:35	慢sql	log	duration: 0.033 ms plan: { "Query Text": "update cron.job_run_details set status = \$1 where runid = \$2", "Plan": { "Nod...	解析
数据库名	5756	2022-06-30 13:48:03	其他	context	SQL statement "update cron.job_run_details set status = \$1 where runid = \$2"	解析
	5727	2022-06-30 13:47:35	慢sql	log	duration: 0.038 ms plan: { "Query Text": "insert into cron.job_run_details (jobid, runid, database, username, command, ...	解析
后端类型	5755	2022-06-30 13:48:03	其他	context	SQL statement "insert into cron.job_run_details (jobid, runid, database, username, command, status) values (\$1,\$2,\$3,...	解析
	5726	2022-06-30 13:47:35	慢sql	log	duration: 0.018 ms plan: { "Query Text": "select timezone from cron.it_job_ext where jobid = 2", "Plan": { "Node Type": ...	解析
	5754	2022-06-30 13:48:03	其他	context	SQL statement "select timezone from cron.it_job_ext where jobid = 2"	解析
	5725	2022-06-30 13:47:35	慢sql	log	duration: 0.038 ms plan: { "Query Text": "update cron.job_run_details set status = \$1, return_message = \$2, end_time = ...	解析
	5753	2022-06-30 13:48:03	其他	context	SQL statement "update cron.job_run_details set status = \$1, return_message = \$2, end_time = \$3 where runid = \$4"	解析
	5724	2022-06-30 13:47:35	慢sql	log	duration: 0.024 ms plan: { "Query Text": "update cron.job_run_details set status = \$1, start_time = \$2 where runid = \$3"...	解析
	5752	2022-06-30 13:48:03	其他	context	SQL statement "update cron.job_run_details set status = \$1, start_time = \$2 where runid = \$3"	解析
	5723	2022-06-30 13:47:35	慢sql	log	duration: 0.046 ms plan: { "Query Text": "update cron.job_run_details set job_pid = \$1, status = \$2 where runid = \$3", "...	解析
	5751	2022-06-30 13:48:03	其他	context	SQL statement "update cron.job_run_details set job_pid = \$1, status = \$2 where runid = \$3"	解析
	5722	2022-06-30 13:47:35	慢sql	log	duration: 0.037 ms plan: { "Query Text": "update cron.job_run_details set status = \$1 where runid = \$2", "Plan": { "Nod...	解析
	5750	2022-06-30 13:48:03	其他	context	SQL statement "update cron.job_run_details set status = \$1 where runid = \$2"	解析
	5127	2022-06-30 13:47:04	慢sql	log	duration: 0.026 ms plan: { "Query Text": "update cron.job_run_details set status = \$1 where runid = \$2", "Plan": { "Nod...	解析
	4934	2022-06-30 13:47:00	其他	context	SQL statement "update cron.job_run_details set status = \$1 where runid = \$2"	解析
	5721	2022-06-30 13:47:35	慢sql	log	duration: 0.034 ms plan: { "Query Text": "insert into cron.job_run_details (jobid, runid, database, username, command, ...	解析
	5749	2022-06-30 13:48:03	其他	context	SQL statement "insert into cron.job_run_details (jobid, runid, database, username, command, status) values (\$1,\$2,\$3,...	解析
	5720	2022-06-30 13:47:35	慢sql	log	duration: 0.017 ms plan: { "Query Text": "select timezone from cron.it_job_ext where jobid = 2", "Plan": { "Node Type": ...	解析
	5126	2022-06-30 13:47:04	慢sql	log	duration: 0.034 ms plan: { "Query Text": "insert into cron.job_run_details (jobid, runid, database, username, command, ...	解析
	5748	2022-06-30 13:48:03	其他	context	SQL statement "select timezone from cron.it_job_ext where jobid = 2"	解析
	4933	2022-06-30 13:47:00	其他	context	SQL statement "insert into cron.job_run_details (jobid, runid, database, username, command, status) values (\$1,\$2,\$3,...	解析
	5125	2022-06-30 13:47:04	慢sql	log	duration: 0.016 ms plan: { "Query Text": "select timezone from cron.it_job_ext where jobid = 2", "Plan": { "Node Type": ...	解析
	4932	2022-06-30 13:47:00	其他	context	SQL statement "select timezone from cron.it_job_ext where jobid = 2"	解析
	5719	2022-06-30 13:47:35	慢sql	log	duration: 0.035 ms plan: { "Query Text": "update cron.job_run_details set status = \$1, return_message = \$2, end_time = ...	解析
	5747	2022-06-30 13:48:03	其他	context	SQL statement "update cron.job_run_details set status = \$1, return_message = \$2, end_time = \$3 where runid = \$4"	解析
	5124	2022-06-30 13:47:04	慢sql	log	duration: 0.035 ms plan: { "Query Text": "update cron.job_run_details set status = \$1, return_message = \$2, end_time = ...	解析
	4931	2022-06-30 13:47:00	其他	context	SQL statement "update cron.job_run_details set status = \$1, return_message = \$2, end_time = \$3 where runid = \$4"	解析

<

首页

数据库日志

5727的执行计划解析器

>

...

执行计划解析器

添加

HTML

SOURCE

QUERY

序列	exclusive	inclusive	rows x	rows	loops	node
1	0	0	↓0.0	0	0	▼ModifyTable job_run_details (const=0..0.01 rows=1 width=196),Operation:Insert
2	0	0	↓0.0	0	0	Result (const=0..0.01 rows=1 width=196)

Planning time : 0 ms

Execution time :0 ms

该查询计划包含 source 数据和该条日志包含的 query 语句

首页

数据库日志

5727的执行计划解析器

执行计划解析器

添加

HTML

SOURCE

QUERY

ModifyTable job_run_details (const=0..0.01 rows=1 width=196),Operation:Insert
Result (const=0..0.01 rows=1 width=196)

首页数据库日志5727的执行计划解析器

执行计划解析器

HTMLSOURCEQUERY

insert into cron.job_run_details (jobid, runid, database, username, command, status) values (\$1,\$2,\$3,\$4,\$5,\$6)

添加

日志时间：可以选择日志时间范围进行搜索，先选择日期；

日志时间2022-01-29 00:00:00 - 2022-01-29 00:00:00

目

最近30秒

最近1分钟

最近10分钟

最近30分钟

最近1小时

最近4小时

最近8小时

最近1天

最近2天

最近7天

<<<>>>

2022年 1月2022年 2月

日	一	二	三	四	五	六	日	一	二	三	四	五	六
26	27	28	29	30	31	1	30	31	1	2	3	4	5
2	3	4	5	6	7	8	6	7	8	9	10	11	12
9	10	11	12	13	14	15	13	14	15	16	17	18	19
16	17	18	19	20	21	22	20	21	22	23	24	25	26
23	24	25	26	27	28	29	27	28	1	2	3	4	5
30	31	1	2	3	4	5	6	7	8	9	10	11	12

选择时间

清空

确定

再选择时间

日志时间

2022-01-29 00:00:00 - 2022-01-29 00:00:00

📅

最近30秒

最近1分钟

最近10分钟

最近30分钟

最近1小时

最近4小时

最近8小时

最近1天

最近2天

最近7天

开始时间

结束时间

00	00	00	00	00	00	00
01	01	01	01	01	01	01
02	02	02	02	02	02	02
03	03	03	03	03	03	03
04	04	04	04	04	04	04
05	05	05	05	05	05	05
06	06	06	06	06	06	06
07	07	07	07	07	07	07
08	08	08	08	08	08	08

选择日期

清空

确定

日志级别：日志级别包括 fatal、notice、info、error 和 log



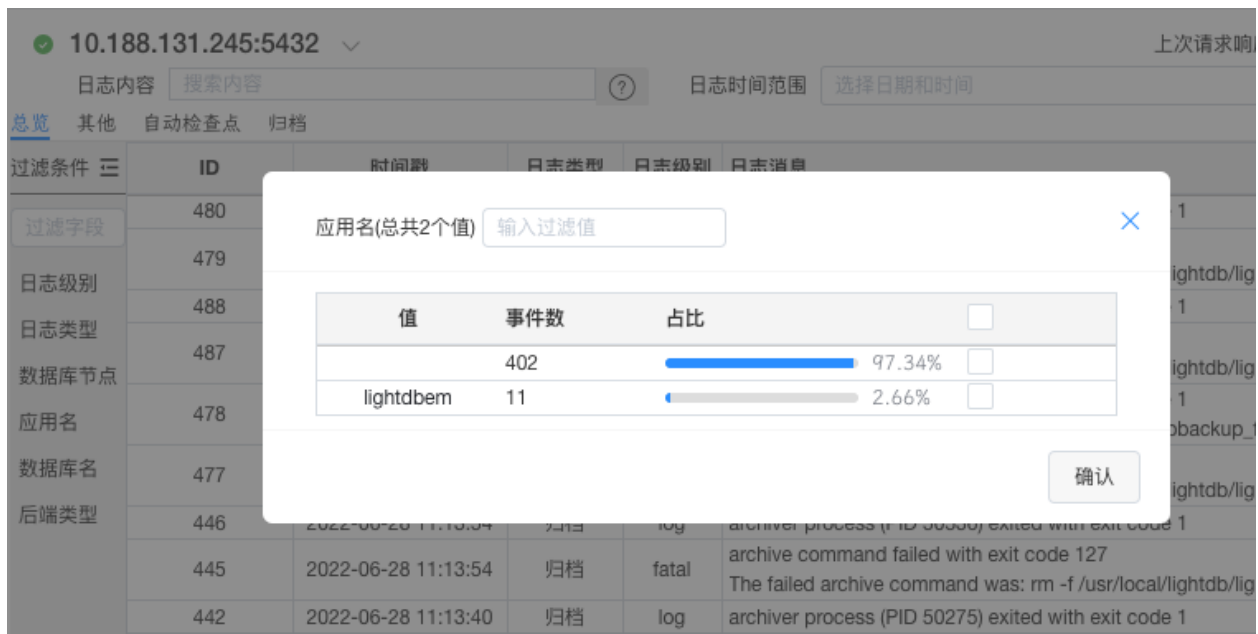
日志类型：日志类型包括临时文件、慢 SQL、其他、auto vacuum 和自动检查点



数据库节点：根据实例 IP 和实例端口进行精确搜索



应用名：连接数据库的应用名称



数据库名：连接的数据库名称



后端类型：后端类型对应视图 `pg_stat_activity` 中的 `backend_type` 列。但是其他类型可能会出现在日志中而不显示在该视图中。

10.188.131.245:5432

上次请

日志内容

搜索内容

?

日志时间范围

选择日期和时间

总览

其他

自动检查点

归档

过滤条件

ID

时间戳

日志类型

日志级别

日志消息

过滤字段

480

479

488

487

478

477

446

445

442

441

2022-06-28 11:13:40

归档

fatal

archive command failed with exit code 127

后端类型(总共4个值)

输入过滤值

值	事件数	占比	
client backend idle	11	2.60%	
in transaction	205	48.46%	
postmaster	2	0.47%	
checkpointer	205	48.46%	
archiver			

确认

日志全屏功能：点击上方按钮可以使当前窗口全屏展示

10.188.132.12:5432

上次请求耗时: 12ms

页面刷新于: 2022-06-29 13:34:33

自动刷新

日志内容

complete

?

日志时间范围

选择日期和时间

按 esc 即可退出全屏模式

保存

总览

其他

自动检查点

归档

过滤条件

ID

时间戳

日志类型

日志级别

日志消息

解析计划

过滤字段

2776

2022-06-28 14:00:31

其他

log

restartpoint complete: wrote 3172 buffers (0.1%); 0 WAL file(s) added, 0 removed, 0 recycled; write=37.892 s, sync=0.003 s, total=37.899 s; sync files=96, longest=0.002 s, average=0.001 s; distance=6763 kB, estimate=6807 kB

2388

2022-06-28 13:30:32

其他

log

restartpoint complete: wrote 3173 buffers (0.1%); 0 WAL file(s) added, 0 removed, 0 recycled; write=38.022 s, sync=0.001 s, total=38.041 s; sync files=95, longest=0.001 s, average=0.001 s; distance=6506 kB, estimate=6812 kB

1982

2022-06-28 13:00:20

其他

log

restartpoint complete: wrote 318 buffers (0.1%); 0 WAL file(s) added, 0 removed, 0 recycled; write=32.422 s, sync=0.003 s, total=32.440 s; sync files=94, longest=0.002 s, average=0.001 s; distance=6846 kB, estimate=6846 kB

共 3 条

50 条/页

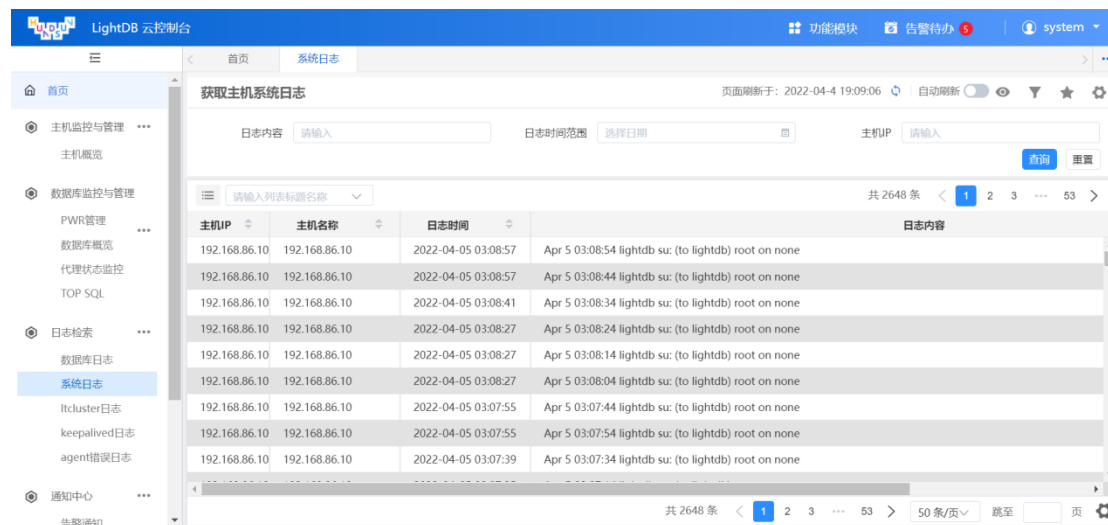
跳至

页

点击右上角的或者 esc 都可以退出全屏状态。

系统日志

系统日志从/var/log/messages 中采集所有日志。



LightDB 云控制台

功能模块 告警待办 5 system

获取主机系统日志

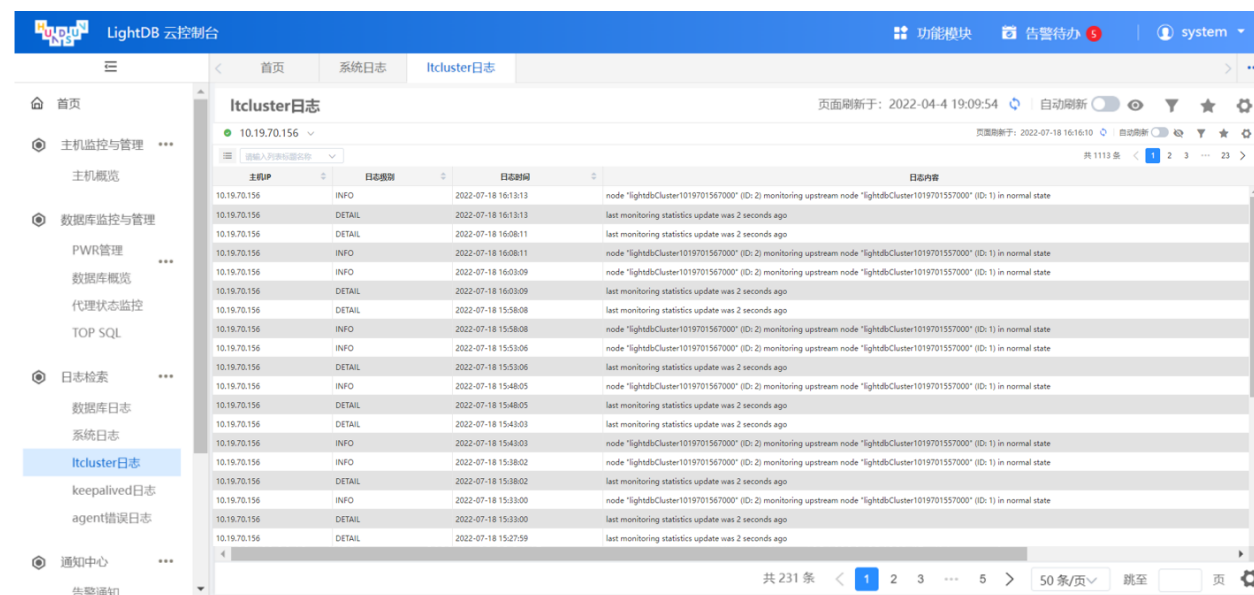
页面刷新于: 2022-04-4 19:09:06 自动刷新 日志内容 请输入 日志时间范围 选择日期 主机IP 请输入 查询 重置

主机IP	主机名称	日志时间	日志内容
192.168.86.10	192.168.86.10	2022-04-05 03:08:57	Apr 5 03:08:54 lightdb su: (to lightdb) root on none
192.168.86.10	192.168.86.10	2022-04-05 03:08:57	Apr 5 03:08:44 lightdb su: (to lightdb) root on none
192.168.86.10	192.168.86.10	2022-04-05 03:08:41	Apr 5 03:08:34 lightdb su: (to lightdb) root on none
192.168.86.10	192.168.86.10	2022-04-05 03:08:27	Apr 5 03:08:24 lightdb su: (to lightdb) root on none
192.168.86.10	192.168.86.10	2022-04-05 03:08:27	Apr 5 03:08:14 lightdb su: (to lightdb) root on none
192.168.86.10	192.168.86.10	2022-04-05 03:08:27	Apr 5 03:08:04 lightdb su: (to lightdb) root on none
192.168.86.10	192.168.86.10	2022-04-05 03:07:55	Apr 5 03:07:44 lightdb su: (to lightdb) root on none
192.168.86.10	192.168.86.10	2022-04-05 03:07:55	Apr 5 03:07:54 lightdb su: (to lightdb) root on none
192.168.86.10	192.168.86.10	2022-04-05 03:07:39	Apr 5 03:07:34 lightdb su: (to lightdb) root on none

共 2648 条 1 2 3 ... 53 50 条/页 跳至 页

Itcluster 日志

Itcluster 日志记录 Itcluster 组件生成的日志，Itcluster 日志只有在集群环境才有。非集群环境看不到 Itcluster 日志菜单。



LightDB 云控制台

功能模块 告警待办 5 system

Itcluster 日志

页面刷新于: 2022-04-4 19:09:54 自动刷新 10.19.70.156 日志内容 请输入 日志时间范围 选择日期 主机IP 请输入 查询 重置

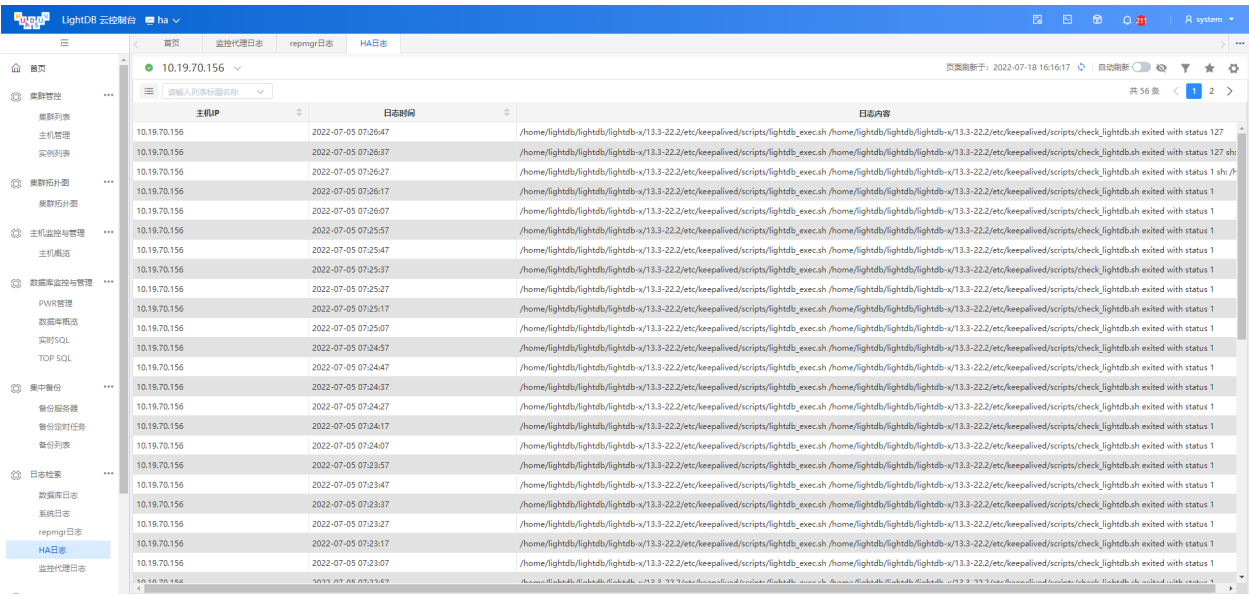
主机IP	日志级别	日志时间	日志内容
10.19.70.156	INFO	2022-07-18 16:13:13	node "lightdbCluster1019701567000" (ID: 2) monitoring upstream node "lightdbCluster1019701557000" (ID: 1) in normal state
10.19.70.156	DETAIL	2022-07-18 16:13:13	last monitoring statistics update was 2 seconds ago
10.19.70.156	DETAIL	2022-07-18 16:08:11	last monitoring statistics update was 2 seconds ago
10.19.70.156	INFO	2022-07-18 16:08:11	node "lightdbCluster1019701567000" (ID: 2) monitoring upstream node "lightdbCluster1019701557000" (ID: 1) in normal state
10.19.70.156	INFO	2022-07-18 16:03:09	node "lightdbCluster1019701567000" (ID: 2) monitoring upstream node "lightdbCluster1019701557000" (ID: 1) in normal state
10.19.70.156	DETAIL	2022-07-18 16:03:09	last monitoring statistics update was 2 seconds ago
10.19.70.156	DETAIL	2022-07-18 15:58:08	last monitoring statistics update was 2 seconds ago
10.19.70.156	INFO	2022-07-18 15:58:08	node "lightdbCluster1019701567000" (ID: 2) monitoring upstream node "lightdbCluster1019701557000" (ID: 1) in normal state
10.19.70.156	INFO	2022-07-18 15:53:06	node "lightdbCluster1019701567000" (ID: 2) monitoring upstream node "lightdbCluster1019701557000" (ID: 1) in normal state
10.19.70.156	DETAIL	2022-07-18 15:53:06	last monitoring statistics update was 2 seconds ago
10.19.70.156	INFO	2022-07-18 15:48:05	node "lightdbCluster1019701567000" (ID: 2) monitoring upstream node "lightdbCluster1019701557000" (ID: 1) in normal state
10.19.70.156	DETAIL	2022-07-18 15:48:05	last monitoring statistics update was 2 seconds ago
10.19.70.156	DETAIL	2022-07-18 15:43:03	last monitoring statistics update was 2 seconds ago
10.19.70.156	INFO	2022-07-18 15:43:03	node "lightdbCluster1019701567000" (ID: 2) monitoring upstream node "lightdbCluster1019701557000" (ID: 1) in normal state
10.19.70.156	INFO	2022-07-18 15:38:02	node "lightdbCluster1019701567000" (ID: 2) monitoring upstream node "lightdbCluster1019701557000" (ID: 1) in normal state
10.19.70.156	DETAIL	2022-07-18 15:38:02	last monitoring statistics update was 2 seconds ago
10.19.70.156	INFO	2022-07-18 15:33:00	node "lightdbCluster1019701567000" (ID: 2) monitoring upstream node "lightdbCluster1019701557000" (ID: 1) in normal state
10.19.70.156	DETAIL	2022-07-18 15:33:00	last monitoring statistics update was 2 seconds ago
10.19.70.156	DETAIL	2022-07-18 15:27:59	last monitoring statistics update was 2 seconds ago

共 231 条 1 2 3 ... 5 50 条/页 跳至 页

Itcluster 用于管理 LightDB 服务器集群中的复制和故障转移。它通过设置备用服务器、监控复制和执行管理任务（如故障转移或手动切换操作）的工具增强了 LightD 的内置热备用功能。Itcluster 为 LightDB 的内置复制机制提供了高级支持，例如通过复制协议的级联复制、时间线切换和基本备份。

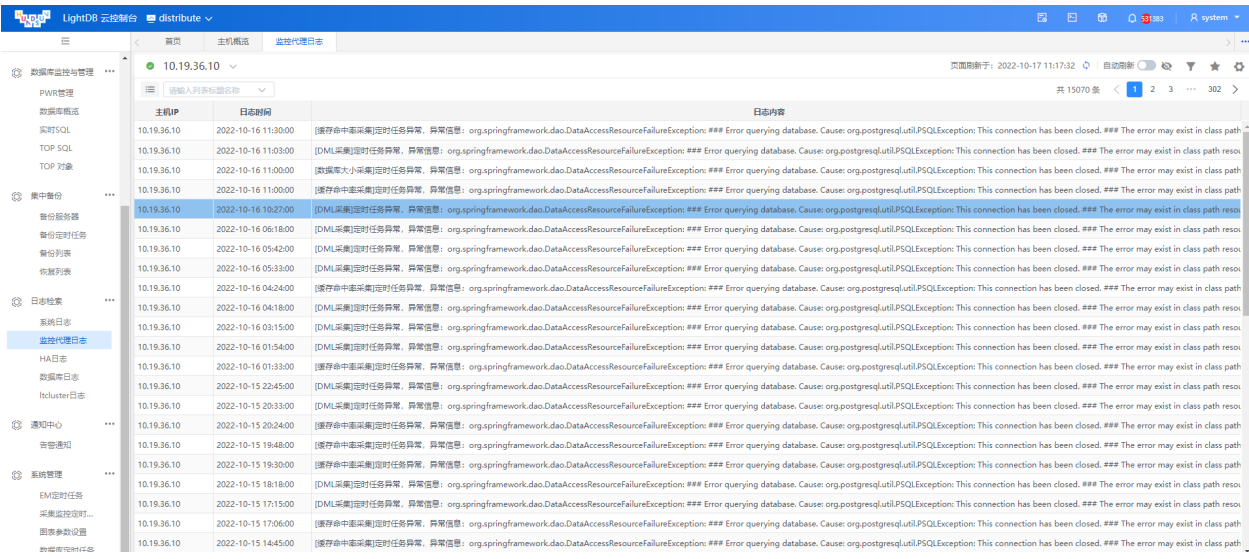
HA 日志

HA 日志记录 keeplived 组件生成的日志，HA 日志只有在集群环境才有。非集群环境看不到 keeplived 日志菜单。



监控代理日志

监控代理日志记录主机的 agent 采集定时任务执行过程中出现的错误。系统捕获 agent 采集定时任务逻辑执行时的异常，记录主机 IP、时间和异常信息 Exception e.getMessage()

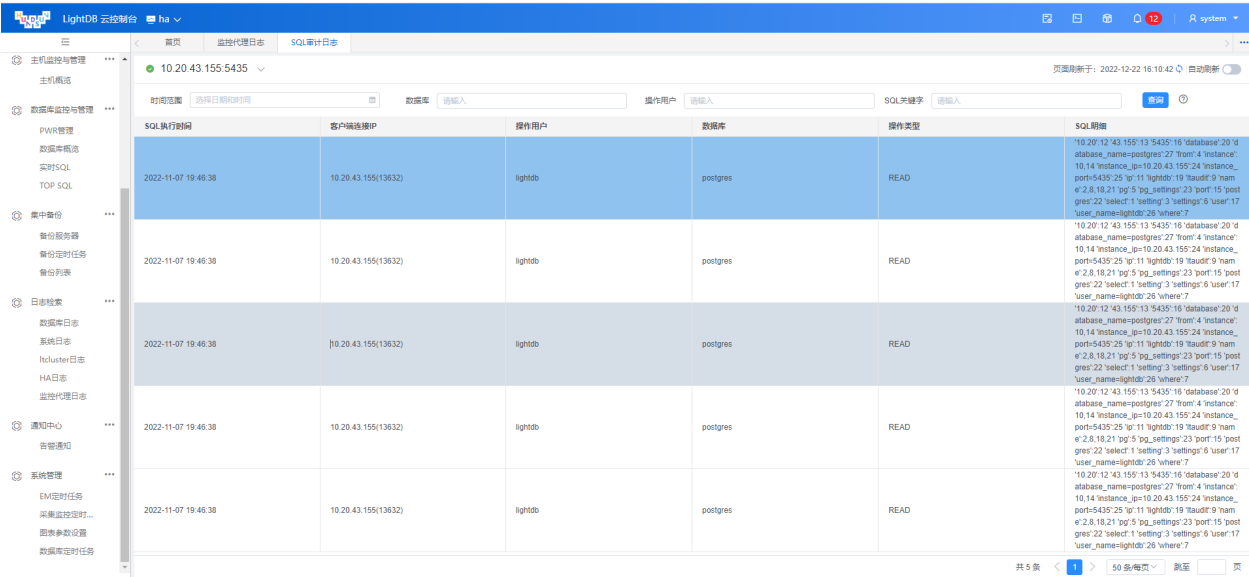


审计日志

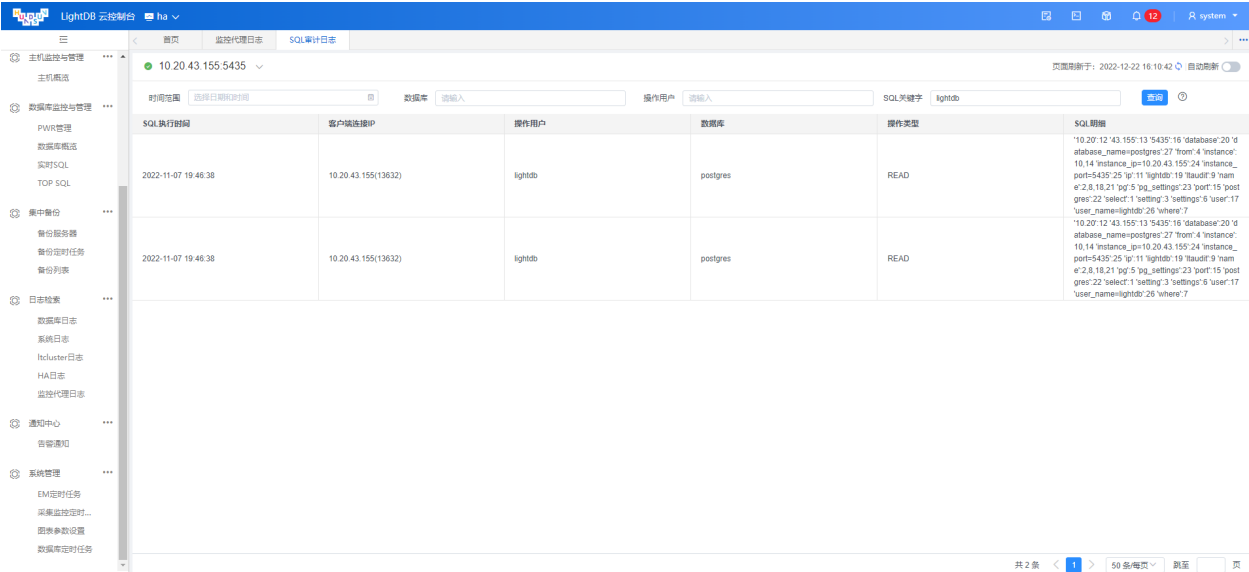
审计是对个人或组织账目的正式检查，通常由独立机构进行。ItAudit 扩展支持为 Lightdb 用户提供生成审计日志的能力，通常需要符合政府、金融或 ISO 认证。生成的审计日志是记录在数据库日志文件中的，分散且不易读。在数据库日志中，如果日志级别冒号后的第一个冒号之前的内容如果是 AUDIT 则是审计日志。

```
// 0 WAL file(s) added, 0 removed, 0 recycled; write=79.047 s; sync=0.001 s; total=79.047 s; sync files=91; longest=0.001 s; average=0.001 s; distance=10321 kB, estimate=10321 kB
2022-10-28 14:54:39.815777T,,,,,checkpoint,,,00000,2022-10-24 12:51:08 CST,0,123912,LOG: checkpoint starting: time
2022-10-28 14:56:17.140024T,,,,,checkpoint,,,00000,2022-10-24 12:51:08 CST,0,123912,LOG: checkpoint complete: wrote 930 buffers (0.7%
); 0 WAL file(s) added, 0 removed, 0 recycled; write=97.319 s; sync=0.001 s; total=97.325 s; sync files=91; longest=0.001 s; average=0.001 s; distance=11797 kB, estimate=11797 kB
2022-10-28 15:17:41.186415T,Navicat,lightdb,postgres,10.188.136.243(49970),client backend,SELECT,00000,2022-10-25 16:41:59 CST,0,129858
,LOG: AUDIT: SESSION,11,1,READ,SELECT,,,select a.*,a.datctype tt from pg_database a",<not logged>
lightdb@ens-10-20-140-128 log$
```

支持用户在 EM 界面查看审计信息进行审计

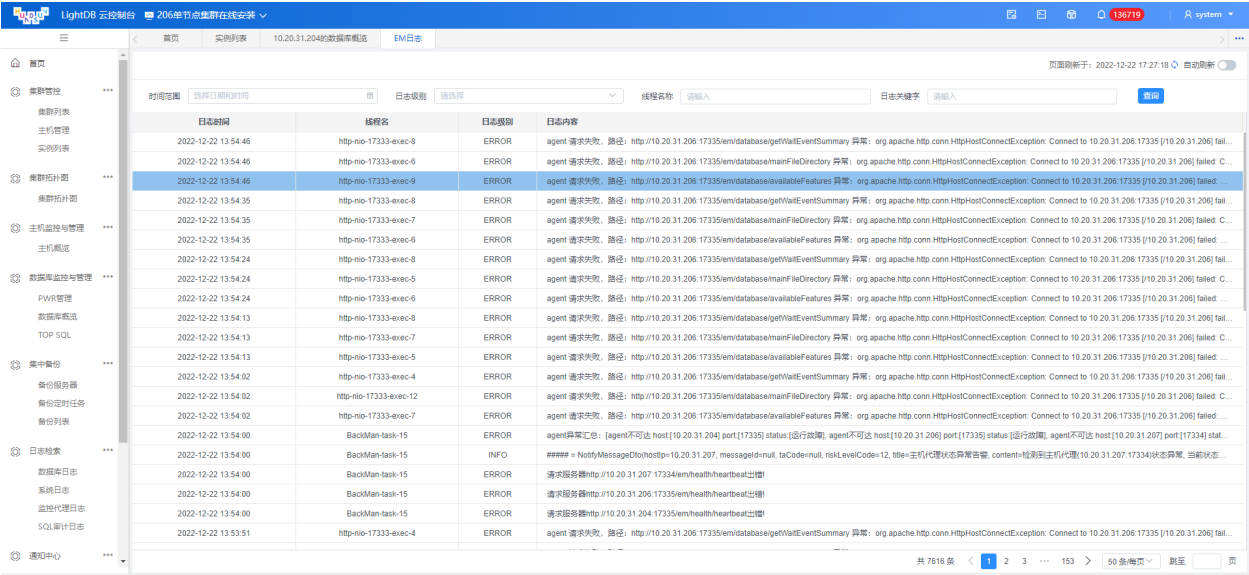


日志搜索查询过滤条件如下：日志搜索：根据关键词进行搜索，关键词实现了全文检索；

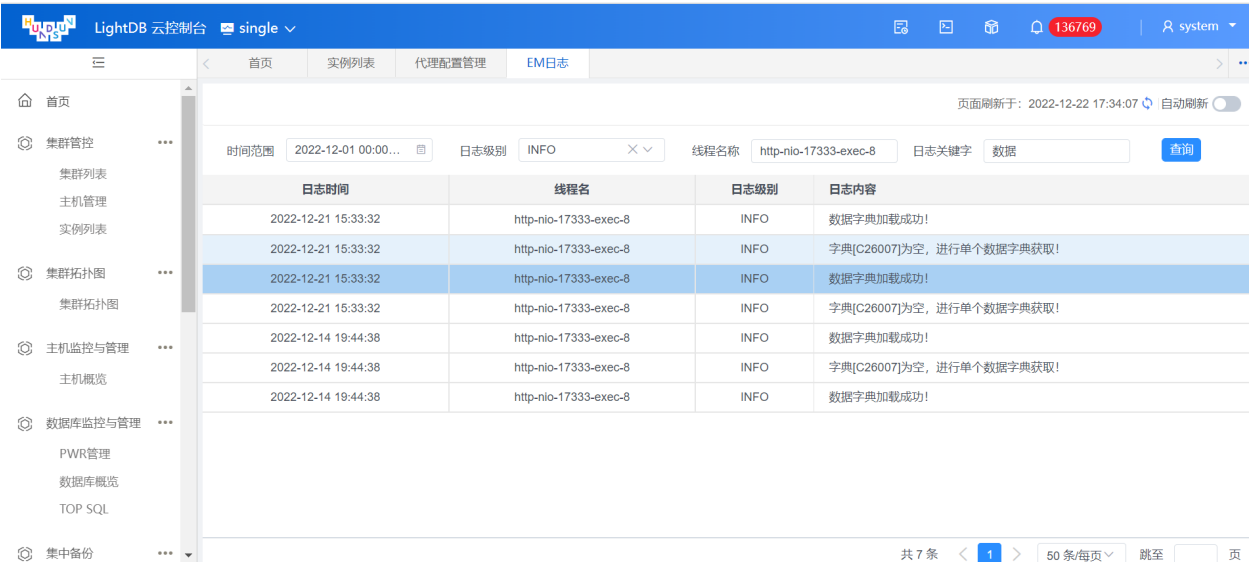


EM 日志

Em 日志是记录我们 em 系统本身的功能，其中记录的日志是我们日志 em 日志中的 lightdb-em.log 的日志。数据库日志采集以行为单位进行读取，以段为单位进行采集。每段的规定是以日期开头到下一个日期开头之前的所有行的内容为一 段。

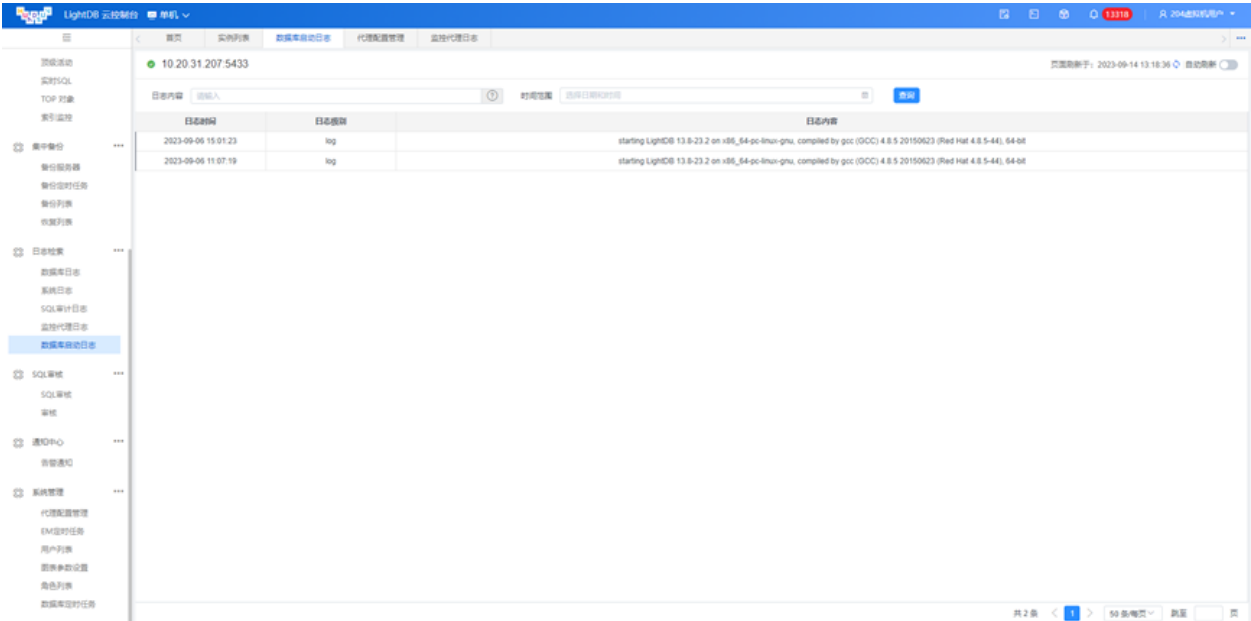


其中支持根据时间范围，日志级别，线程名称，日志关键字查找日志，



数据库启动日志

数据库启动日志采集以行为单位进行读取，以段为单位进行采集。每段的规定是以日期开头到下一个日期开头之前的所有行的内容为一。在数据库日志中，如果日志内容是以 starting LightDB 开头则是数据库启动日志。



3.12 统一 sql 代理

双方比对报告

该界面展示监控到的变动 sql 和对应表的数据变动，可以按照 ip、traceId、功能号进行查询。

下载 em

- 1. 访问 [em 下载页面](#) 下载对应平台的 em 版本。
- 2. em 版本要求 >=23.4
- 3. em 安装手册参考 [em 在线安装手册](#) 查看。

进入数据校准页面前准备

小技巧：默认安装的 em 不会展示数据同步页面，在 em 的 配置库 em 中执行如下 sql 来显示 双发比对页面 和 数据较准同步页面之后，需再 重启 em 生效。

```
# 登录em所在服务器，使用命令行工具ltsql连接到em的配置库em，执行下面sql
[lightdb@node-3 ~]$ ltsql -p 5434 -d em
ltsql (13.8-22.4)
Type "help" for help.
lightdb@em=#

-- 删除双发比对报告和 数据校准同步菜单
delete from sys_menu where id in (800,801,810,820);
```

(续下页)

(接上页)

```
INSERT INTO sys_menu
(tenant_code, parent_id, refer_id, menu_code, menu_name, url, order_num, resource_
→code, query_string, url_prefix, menu_level, menu_style, id, auto_auth, ta_code,
→verify_enable, hide_component)
VALUES ('*', 1, 1, 'unisqlAgent', '统一SQL代理', '#', 7, '', NULL, NULL, 0, NULL, 800, 0, '*', '0', '
→'),
    ('*', 800, 1, 'compareServiceTab', '监控比对服务', '#', 1, '', NULL, NULL, 0, NULL, 801,
→0, '*', '0', ' '),
    ('*', 801, 1, 'compareReport', '双发比对报告', '/tasystem/emPage/unisqlAgent/
→compareReport', 1, '', NULL, NULL, 0, NULL, 810, 0, '*', '0', ' '),
    ('*', 801, 1, 'dataValidSync', '数据校准同步', '/tasystem/emPage/unisqlAgent/
→dataValidSync', 2, '', NULL, NULL, 0, NULL, 820, 0, '*', '0', ' ');
-- 收藏的菜单
delete from sys_user_menu where (user_id, menu_id) in ((1,810),(1,820));
select setval('sys_user_menu_id_seq',(select max(id) from sys_user_menu));
INSERT INTO sys_user_menu (tenant_code, user_id, menu_id, menu_order)
VALUES ('*', 1, 810, 2),
    ('*', 1, 820, 3);
-- 角色的菜单
DELETE FROM sys_role_menu WHERE (role_id, menu_id) in ((1, 800),(1, 810),(1, 820));
INSERT INTO sys_role_menu (tenant_code, role_id, menu_id, menu_order)
VALUES ('*', 1, 800, 1),
    ('*', 1, 810, 2),
    ('*', 1, 820, 3);
-- 开放SQL审核下的数据库连接信息页面
update sys_menu set refer_id=1 where id in (115,116,119);
```

```
# 比对服务配置项说明，修改完这些配置项需要重启``em``
# 修改em工作目录下config/jrescloud.properties这个配置文件，添加对比服务数据源
dyn.spring.datasources[1].name=comparison
dyn.spring.datasources[1].driverClassName=org.postgresql.Driver
dyn.spring.datasources[1].url=jdbc:postgresql://${em_host}:${port}/unisql_comparison?
→ApplicationName=lightdbem&socketTimeout=5000&options=-c%20lock_timeout=30min
dyn.spring.datasources[1].username=lightdb
dyn.spring.datasources[1].password=${password}
dyn.spring.datasources[1].maxActive=208
dyn.spring.datasources[1].maxWait=5000
dyn.spring.datasources[1].maxIdle=1
dyn.spring.datasources[1].removeAbandoned=true
dyn.spring.datasources[1].removeAbandonedTimeout=600
dyn.spring.datasources[1].logAbandoned=true

# 对比服务数据源配置，变量解释说明
${em_host}:em所在服务器的ip地址
${port}:数据库实例lightdb的监听端口，默认就是5434
${password}:数据库用户lightdb的密码
```

备注：比对服务数据源，ip，数据库用户名，数据库用户密码与em的配置库 em一致。

```
# 修改完这些配置项需要重启``em``
# 修改em工作目录下config/jrescloud.properties这个配置文件
# 添加pulsar客户端消费者配置
sqlmonitor.dbType=oracle
sqlmonitor.pulsarServiceUrl=pulsar://${pulsar_ip}:${pulsar_port}
```

(续下页)

(接上页)

```
sqlmonitor.topicName=persistent://public/default/recordMonitor
sqlmonitor.subscriptionName=sql-subscription
# 变量说明
${pulsar_ip}:pulsar服务器ip, 与em所在服务器ip一样
${pulsar_port}:pulsar服务器监听端口, 默认端口6650
```

备注: pulsar 客户端消费者配置, 配置项 `sqlmonitor.pulsarServiceUrl` 表示 pulsar 服务端信息, 其中 ip 与 em 的 ip 一致, 端口默认 6650。

```
# sql驱动拦截微服务全量sql的-D参数配置
-Dsqlmonitor.interceptAllSql=true

# 在log4j2的配置文件log4j2.xml中配置拦截微服务全量sql的``logger``和``appender``配置
<!-- 微服务工作目录下logs目录下sqlMonitor.log文件 -->
<RollingFile name="sqlMonitorFile" fileName="logs/sqlMonitor.log" filePattern="logs/${
→{date:yyyy-MM}/sqlMonitor-%d{yyyy-MM-dd}-%i.log.gz">
    <PatternLayout pattern="%d{yyyy-MM-dd HH:mm:ss.SSS} [%t] %-5level %logger{36}%n"
→- %msg%n"/>
    <Policies>
        <TimeBasedTriggeringPolicy/>
        <SizeBasedTriggeringPolicy size="250 MB"/>
    </Policies>
    <DefaultRolloverStrategy max="10"/>
</RollingFile>

<Logger name="com.hundsun.lightdb.monitor.proxy.jdbc.MonitorPreparedStatement" level=
→"info">
    <AppenderRef ref="sqlMonitorFile"/>
</Logger>
```

```
# 比对服务配置项说明, 修改完这些配置项需要重启``em``
# 修改em工作目录下config/jrescloud.properties这个配置文件
# 配置项``sqlmonitor.ignoredFields``表示``忽略表字段``, 格式为``表名.
→字段名``, 多个忽略的表字段之间英文逗号","分隔, 举例如下:
sqlmonitor.ignoredFields=bps_acpt_tpl.UPDATE_TIME,bps_acpt_tpl.UPDATE_DATE

# 配置项``sqlmonitor.
→comparisonRuleFilePath``表示``按照比较规则比较字段``, 该配置项内容为比较规则文件在服务器完整路径,
sqlmonitor.comparisonRuleFilePath=/home/lightdb/compare.xlsx
```

```
# 启动em过程中先启动pulsar, 具体执行命令如下
# 进入em工作目录
[lightdb@node-3 scripts]$ cd /data/lightdb/em233/LightDB1.0-em-V202303-00-000-e17-x86_
→64/em
# 进入em脚本目录
[lightdb@node-3 em]$ cd scripts/
# 执行启动脚本, 优先启动pulsar, 再启动em进程
[lightdb@node-3 scripts]$ sh start_all.sh --start_pulsar
doing start standalone ...
starting standalone, logging to /data/lightdb/em233/LightDB1.0-em-V202303-00-000-e17-
→x86_64/em/pulsar/logs/pulsar-standalone-node-3.log
Note: Set immediateFlush to true in conf/log4j2.yaml will guarantee the logging event.
→is flushing to disk immediately. The default behavior is switched off due to
→performance considerations.
```

(续下页)

(接上页)

```
log path: /data/lightdb/em233/LightDB1.0-em-V202303-00-000-el7-x86_64/em/redis/redis.
↪log
11:22:19 em_start.sh kill old em process and green lightdb instance process first
11:22:20 em_stop.sh green LTBASE: [/data/lightdb/em233/LightDB1.0-em-V202303-00-000-
↪el7-x86_64/lightdb-x-for-em]
11:22:20 em_stop.sh green LTHOME: [/data/lightdb/em233/LightDB1.0-em-V202303-00-000-
↪el7-x86_64/lightdb-x-for-em/13.8-22.4]
11:22:20 em_stop.sh green LTDATA: [/data/lightdb/em233/LightDB1.0-em-V202303-00-000-
↪el7-x86_64/lightdb-x-for-em/13.8-22.4/data]
11:22:20 em_stop.sh No green lightdb instance process !!!
11:22:20 em_start.sh GREEN_LIGHTDB_VERSION : 13.8-22.4
11:22:20 em_start.sh LTHOME: [/data/lightdb/em233/LightDB1.0-em-V202303-00-000-el7-
↪x86_64/lightdb-x-for-em/13.8-22.4]
11:22:20 em_start.sh LTDATA: [/data/lightdb/em233/LightDB1.0-em-V202303-00-000-el7-
↪x86_64/lightdb-x-for-em/13.8-22.4/data]
11:22:20 em_start.sh checking green lightdb instance process whether started or not...
↪.
11:22:20 em_start.sh LTDATA :/data/lightdb/em233/LightDB1.0-em-V202303-00-000-el7-x86_
↪64/lightdb-x-for-em/13.8-22.4/data] not occupy. now start green lightdb instance
waiting for server to start....2023-12-14 11:22:20.515377T,,,,postmaster,,00000,2023-
↪12-14 11:22:20 CST,0,20300,LOG: requesting named LWLockTranch for canopy_query_
↪stats
2023-12-14 11:22:20.569655T,,,,postmaster,,00000,2023-12-14 11:22:20 CST,0,20300,
↪LOG: LightDB autoprewarm: prewarm dbnum=0
2023-12-14 11:22:20.577267T,,,,postmaster,,00000,2023-12-14 11:22:20 CST,0,20300,
↪INFO: invoking IpcMemoryCreate(size=440696832)
2023-12-14 11:22:20.882279T,,,,postmaster,,00000,2023-12-14 11:22:20 CST,0,20300,
↪LOG: redirecting log output to logging collector process
2023-12-14 11:22:20.882279T,,,,postmaster,,00000,2023-12-14 11:22:20 CST,0,20300,
↪HINT: Future log output will appear in directory "log".
done
server started
11:22:20 em_start.sh start green lightdb instance sucess and prepare start em process
11:22:20 em_start.sh now start em process...
11:22:20 em_start.sh em log path: /data/lightdb/em233/LightDB1.0-em-V202303-00-000-
↪el7-x86_64/em/logs
/data/lightdb/em233/LightDB1.0-em-V202303-00-000-el7-x86_64/em/scripts
/data/lightdb/em233/LightDB1.0-em-V202303-00-000-el7-x86_64/em/scripts
log path: /data/lightdb/em233/LightDB1.0-em-V202303-00-000-el7-x86_64/em/gotty/gotty.
↪log
```

警告:

1. 启动 pulsar 进程，默认占用 jvm 内存大小 -Xms2g -Xmx2g -XX:MaxDirectMemorySize=4g，至少占用服务器内存 2 个 g。
2. 为避免因服务器物理内存不足造成 em 进程启动失败，在 em 家目录下 pulsar/scripts 目录下执行脚本 set_pulsar_mem.sh 降低启用 pulsar 进程所需 jvm 内存。

执行set_pulsar_mem.sh脚本来的步骤如下：

```
[lightdb@node-3 ~]$ cd /data/lightdb/em233/LightDB1.0-em-V202303-00-000-el7-x86_64/
↪em
[lightdb@node-3 em]$ cd pulsar/scripts/
[lightdb@node-3 scripts]$ sh set_pulsar_mem.sh
PULSAR_MEM has been set in ~/.bashrc and sourced.
```

备注： 启动 em 进程的 java 命令，需要额外补充 spring 命令行参数 `--spring.config.location` 引| 用到 em 配置文件 `jrescloud.properties`，举例 `--spring.config.location=/data/lightdb/em233/LightDB1.0-em-V202303-00-000-el7-x86_64/em/config/jrescloud.properties`

进入数据比对服务结果查询页面

1. 进入 em 导航栏 统一 sql 代理 -> 双发比对报告页面
2. 查询条件 Trace ID，全链路日志标识。
3. 查询条件 IP，微服务 ip。
4. 查询条件 Function ID，微服务功能号，比如 `bps.deleteNoticeInfo`

Trace ID: 7070d137c8440a1083e612b3e773d1d
源数据库: ORACLE 目标数据库: ob

表名: bps_notice_info

SQL:
delete from bps_notice_info where position_str=?

CURR_TIME	POSITION_STR	UP_FLAG	TOP_POS	SERIAL_NO	UPDATE_DATE	NOTICE_CAPTION	CURR_DATE	INIT_DATE	UPDATE_TIME	REMARK_WAY	PUSH_DATE	OP_BRANCH_NO	NOTICE_CODE
ORACLE-44404	123123		1	ORACLE-44404	20231201	123123123	20231201	20231201	ORACLE-44404	ob		-1	123123123
ob-110113				ob-110113					ob-110113	ob			

备注： 比对结果报告说明

1. traceid 全链路日志 id
2. 源数据库，目标数据库
3. 引起表数据变动 sql 影响到的表名
4. 引起表数据变动的 sql（预编译格式）
5. 表格形式展示表数据变化的多行记录，其中数据变化的列的值标注颜色

小技巧：

停止 em 过程中，先停止 em 进程，最后再停止 pulsar 服务

```
# 在em脚本目录(em工作目录/scripts)执行停止脚本，先停止em进程，再停止pulsar
[lightdb@node-3 scripts]$ sh stop_all.sh --stop_pulsar
/data/lightdb/em233/LightDB1.0-em-V202303-00-000-el7-x86_64/em/scripts
kill em 20419 process success!
11:25:14 em_stop.sh green LTBASE: [/data/lightdb/em233/LightDB1.0-em-V202303-00-000-
→el7-x86_64/lightdb-x-for-em]
11:25:14 em_stop.sh green LTHOME: [/data/lightdb/em233/LightDB1.0-em-V202303-00-000-
→el7-x86_64/lightdb-x-for-em/13.8-22.4]
```

(续下页)

(接上页)

```
11:25:14 em_stop.sh green LTDATA: [/data/lightdb/em233/LightDB1.0-em-V202303-00-000-
→el7-x86_64/lightdb-x-for-em/13.8-22.4/data]
waiting for server to shut down.... done
server stopped
Stop redis server redis_port=[18331] success!
kill gotty 20581 process success!
# 开始停止pulsar服务
doing stop standalone ...
stopping standalone
Shutdown is in progress... Please wait...
Shutdown is in progress... Please wait...
Shutdown is in progress... Please wait...
Shutdown is in progress... Please wait...
Shutdown is in progress... Please wait...
Shutdown is in progress... Please wait...
Shutdown is in progress... Please wait...
Shutdown is in progress... Please wait...
Shutdown is in progress... Please wait...
Shutdown completed.
```

数据校准同步

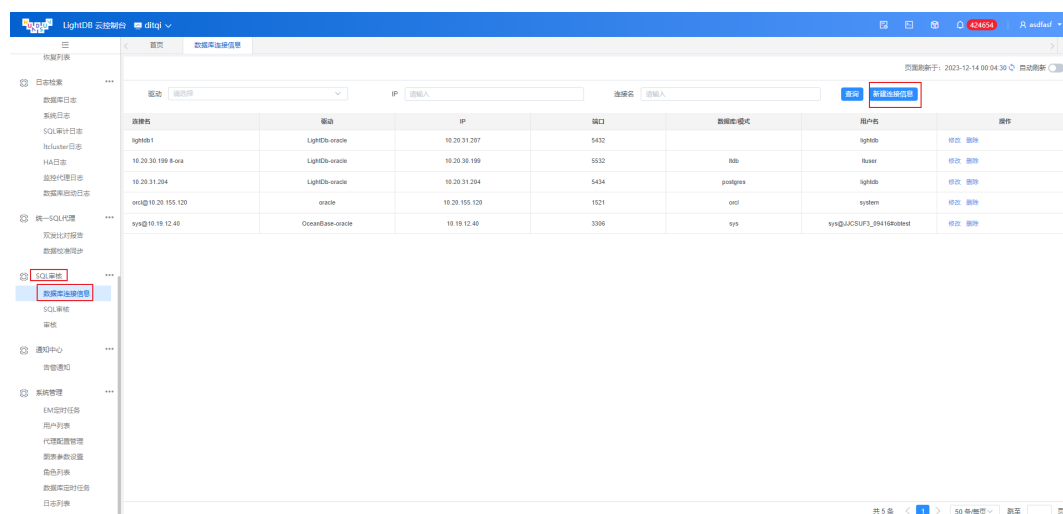
在交互界面，用户自己写 sql 语句 (select 的)，能自动以 oracle 为准，同步对应记录到 ob 等信创库。

前期准备

1. 访问 [em 下载页面](#) 下载对应平台的 em 版本。
2. em 版本要求 ≥ 23.4
3. em 安装手册参考 [em 在线安装手册](#) 查看。
4. 数据同步工具 etl 参考 [数据同步 etl 在线手册](#) 查看。

在 em 页面新增数据库连接

1. 前提已安装 em 并启动成功。
2. 点击 em 导航栏中的 sql 审核 -> 数据库连接信息 -> 新建连接信息。



3. 在新建数据库连接的弹出窗口 添加数据库连接：
- A. 选择驱动，输入 IP, 端口，数据库，用户名，密码，点击 测试数据库连接
 - B. 点击 确定完成 新建数据库连接

* 驱动

oracle

X v

* 连接名

orcl@10.20.155.120

* IP

10.20.155.120

* 端口

1521

* 数据库/模式

orcl

* 用户名

system

* 密码

✓ 测试数据库连接

取消

确定

进入数据校准页面前准备

警告：默认安装的 em 不会展示数据同步页面，在 em 的 配置库 em 执行如下 sql 来显示 数据同步页面，需再 重启 em 生效。

```
delete from sys_menu where id in (800,801,810,820);
INSERT INTO sys_menu
(tenant_code, parent_id, refer_id, menu_code, menu_name, url, order_num, resource_
↪code, query_string, url_prefix, menu_level, menu_style, id, auto_auth, ta_code,
↪verify_enable, hide_component)
VALUES ('*',1,1,'unisqlAgent','统一SQL代理','#',7,'',NULL,NULL,0,NULL,800,0,'*', '0', '
↪'),
        ('*',800,1,'compareServiceTab','监控比对服务','#',1,'',NULL,NULL,0,NULL,801,
↪0,'*', '0', ' '),
        ('*',801,1,'compareReport','双发比对报告','/tasystem/emPage/unisqlAgent/
↪compareReport',1,'',NULL,NULL,0,NULL,810,0,'*', '0', ' '),
        ('*',801,1,'dataValidSync','数据校准同步','/tasystem/emPage/unisqlAgent/
↪dataValidSync',2,'',NULL,NULL,0,NULL,820,0,'*', '0', ' ');
-- 收藏的菜单
delete from sys_user_menu where (user_id, menu_id) in ((1,810),(1,820));
select setval('sys_user_menu_id_seq',(select max(id) from sys_user_menu));
INSERT INTO sys_user_menu (tenant_code, user_id, menu_id, menu_order)
```

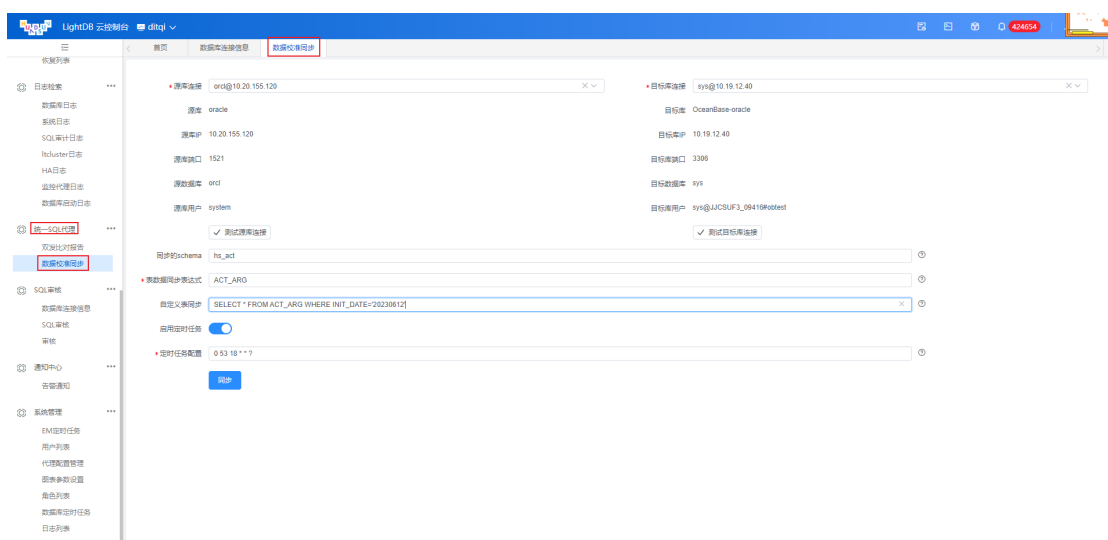
(续下页)

(接上页)

```
VALUES ('*', 1, 810, 2),
        ('*', 1, 820, 3);
-- 角色的菜单
DELETE FROM sys_role_menu WHERE (role_id, menu_id) in ((1, 800), (1, 810), (1, 820));
INSERT INTO sys_role_menu (tenant_code, role_id, menu_id, menu_order)
VALUES ('*', 1, 800, 1),
        ('*', 1, 810, 2),
        ('*', 1, 820, 3);
-- 开放SQL审核下的数据库连接信息页面
update sys_menu set refer_id=1 where id in (115,116,119);
```

进入数据校准页面

1. 进入 em 导航栏 统一 sql 代理 -> 数据校准同步页面
2. 选择 同步的 schema，支持多个 schema 同步，多个 schema 之间英文逗号，分隔。
3. 选择 表数据同步表达式，schema 下表名，支持正则表达式
4. 选择 自定义表同步，一个查询 sql 只支持一张表，多个查询直接用英文分号分隔，样例数据 `SELECT * FROM ACT_ARG WHERE INIT_DATE=' 20230612'`。同步逻辑：根据配置的 sql 删除目标库中对应的数据，如果根据条件能查询到源数据库中的数据则同步到目标表。
5. 选择 定时任务配置，支持 java cron 表达式，后台启动数据同步 java 进程并定时执行数据同步。



可以在 em 工作目录下查看数据迁移详细日志

```
# 进入em的家目录
[lightdb@node-3 ~]$ cd /data/lightdb/em233/LightDB1.0-em-V202303-00-000-e17-x86_64/em
[lightdb@node-3 em]$
[lightdb@node-3 em]$
# 进入数据同步进程执行的日志目录
[lightdb@node-3 em]$ cd lightdb_etl/logs/lightdb-etl/
[lightdb@node-3 lightdb-etl]$
[lightdb@node-3 lightdb-etl]$
# 查看数据同步进程的日志内容
[lightdb@node-3 lightdb-etl]$ tail -fn500 lightdb-etl.log
2023-12-14 00:43:11.840 [background-preinit] INFO org.hibernate.validator.internal.
```

(续下页)

(接上页)

```
→util.Version - HV000001: Hibernate Validator 6.1.7.Final
2023-12-14 00:43:12.000 [main] INFO com.hundsun.lightdb.
→ProcedureTransformerApplication - Starting ProcedureTransformerApplication on node-
→3 with PID 22955 (/data/lightdb/em233/LightDB1.0-em-V202303-00-000-e17-x86_64/em/
→lightdb_etl/lightdb-etl-23.3.1.0-beta.jar started by lightdb in /data/lightdb/em233/
→LightDB1.0-em-V202303-00-000-e17-x86_64/em/lightdb_etl)
2023-12-14 00:43:12.004 [main] INFO com.hundsun.lightdb.
→ProcedureTransformerApplication - No active profile set, falling back to default
→profiles: default
2023-12-14 00:43:13.408 [main] INFO org.springframework.boot.web.embedded.tomcat.
→TomcatWebServer - Tomcat initialized with port(s): 12301 (http)
2023-12-14 00:43:13.433 [main] INFO org.apache.coyote.http11.Http11NioProtocol -
→Initializing ProtocolHandler ["http-nio-12301"]
2023-12-14 00:43:13.435 [main] INFO org.apache.catalina.core.StandardService -
→Starting service [Tomcat]
2023-12-14 00:43:13.436 [main] INFO org.apache.catalina.core.StandardEngine -
→Starting Servlet engine: [Apache Tomcat/9.0.76]
2023-12-14 00:43:13.583 [main] INFO org.apache.catalina.core.ContainerBase.[Tomcat].
→[localhost].[/] - Initializing Spring embedded WebApplicationContext
2023-12-14 00:43:13.583 [main] INFO org.springframework.boot.web.servlet.context.
→ServletWebServerApplicationContext - Root WebApplicationContext: initialization
→completed in 1503 ms
2023-12-14 00:43:13.882 [main] INFO org.springframework.scheduling.concurrent.
→ThreadPoolTaskExecutor - Initializing ExecutorService 'applicationTaskExecutor'
2023-12-14 00:43:14.061 [main] INFO org.springframework.scheduling.concurrent.
→ThreadPoolTaskScheduler - Initializing ExecutorService 'taskScheduler'
2023-12-14 00:43:14.093 [main] INFO org.apache.coyote.http11.Http11NioProtocol -
→Starting ProtocolHandler ["http-nio-12301"]
2023-12-14 00:43:14.143 [main] INFO org.springframework.boot.web.embedded.tomcat.
→TomcatWebServer - Tomcat started on port(s): 12301 (http) with context path ''
2023-12-14 00:43:14.148 [main] INFO com.hundsun.lightdb.
→ProcedureTransformerApplication - Started ProcedureTransformerApplication in 2.772
→seconds (JVM running for 5.12)
2023-12-14 00:43:14.210 [main] INFO com.hundsun.lightdb.core.EtlGlobalCtx - enable
→functions ["transfertabledata"]
```

3.13 通知中心

通知中心目前展示告警待办发送的告警通知列表。

告警通知

告警通知主要展示的是主机或数据库超过配置阈值，系统给用户发送待办告警的记录信息

告警通知

告警代码

请输入

通知方式

请选择

查询

重置

请输入列表标题名称

共 1772 条

123...36

告警代码	告警名称	通知信息	通知方式	消息结果	发送时间
1	cpu总使用率告警	当前主机 10.19.69.193,cpu总使用率 80.21%	站内通知	已发送	2022-01-24 00:05:31
1	cpu总使用率告警	当前主机 10.19.69.193,cpu总使用率 80.36%	站内通知	已发送	2022-01-24 00:05:01
1	cpu总使用率告警	当前主机 10.19.69.193,cpu总使用率 81.66%	站内通知	已发送	2022-01-24 00:04:31
1	cpu总使用率告警	当前主机 10.19.69.193,cpu总使用率 82.79%	站内通知	已发送	2022-01-24 00:04:01
1	cpu总使用率告警	当前主机 10.19.69.193,cpu总使用率 81.22%	站内通知	已发送	2022-01-24 00:03:01
1	cpu总使用率告警	当前主机 10.19.69.193,cpu总使用率 83.29%	站内通知	已发送	2022-01-24 00:02:31
1	cpu总使用率告警	当前主机 10.19.69.193,cpu总使用率 84.35%	站内通知	已发送	2022-01-24 00:01:31
1	cpu总使用率告警	当前主机 10.19.69.193,cpu总使用率 95.99%	站内通知	已发送	2022-01-24 00:00:01
1	cpu总使用率告警	当前主机 10.19.69.193,cpu总使用率 80.88%	站内通知	已发送	2022-01-23 23:59:31
1	cpu总使用率告警	当前主机 10.19.69.193,cpu总使用率 82.49%	站内通知	已发送	2022-01-23 23:57:01

共 1772 条

123...36

50 条/页

跳至

页

主机和数据库告警阈值配置表：

告警名称	用途	监控间隔	阈值	备注
cpu 总使用率告警	cpu 总使用率超过 80%	根据数据采集周期而定，实时	cpu_total_usage>80	lem_host_cpz 表 cpu_total_usage 字段
磁盘使用率告警	磁盘使用率告警超过 80%	同上	disk_usage>80	lem_host_diskfiles system 表 usage 字段
内存使用率告警	内存使用率告警超过 80%	同上	memory_usage_percent>80	lem_host_memory 表 memory_usage_percent 字段
日志告警		同上	db_log_level=error,fatal	lem_db_log 表 db_log_level 字段
数据库失效对象告警		同上		暂未实现
剩余冻结时间 (vacuum) 告警		同上	percent_towards_emergency_autovac>90	lem_db_age 表 percent_towards_emergency_autovac 字段
数据库内置定时任务告警	数据库定时任务告警	同上	status=failed	(select * from cron.job_run_details where status='failed')
data 目录可用空间不足告警	data 目录可用空间不足告警	同上	free_space<10,free_space_percent<10	

备份服务器备份目录可用空间不足告警	备份服务器备份目录可用空间不足告警	同上	free_space<10,free_space_percent<10	
ltcluster 状态异常告警	ltcluster 状态异常告警	同上		
pwr20 分钟内没有数据产生告警	pwr20 分钟内没有数据产生告警	同上		
数据库节点状态异常告警	数据库节点状态异常告警	同上		
定时清理任务异常	定时清理任务异常告警	同上		

告警				
备份失败告警	备份失败告警	同上		
恢复失败告警	恢复失败告警	同上		
数据库日志解析异常告警	数据库日志解析异常告警	同上		

3.14 系统管理

系统管理模块主要包括 LightDB EM 全局设置的功能。

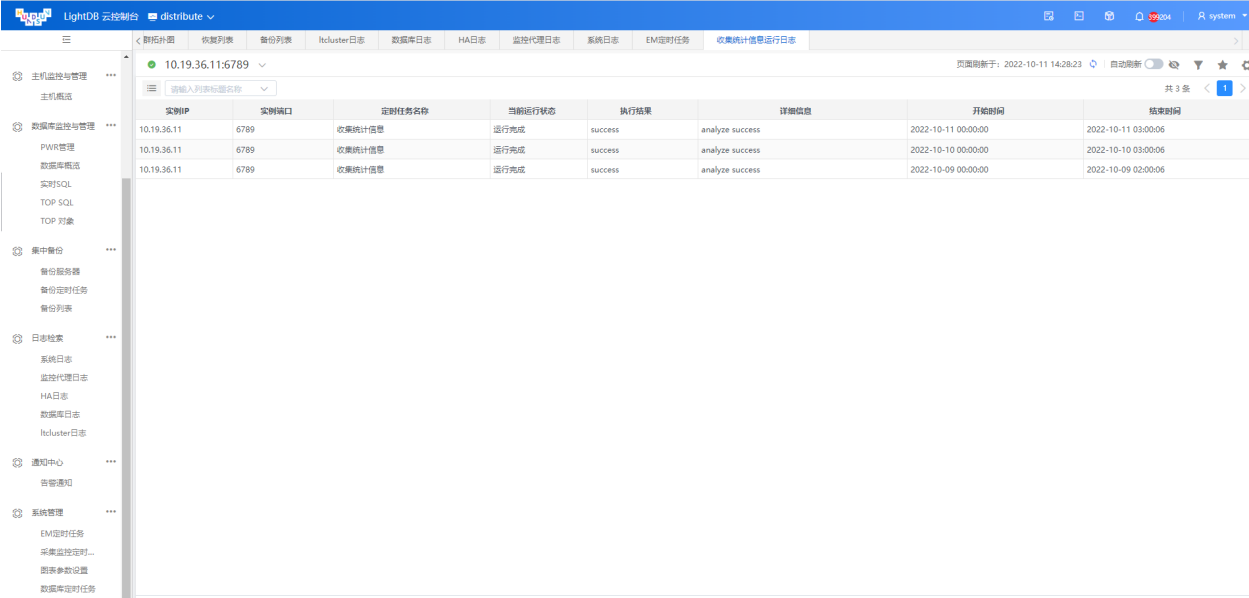
EM 定时任务

EM 定时任务配置的是采集和数据库定时任务外的其它定时任务。目前定时任务有获取最新的消息通知、收集统计信息、定时 vacuum、定时获取 agent 心跳和定时备份任务。

定时任务	执行周期（cron 表达式）	是否启用 (1: 是; 0: 否)
收集统计信息	0 0 1 * * ?	1
定时 vacuum	0 0 1 * * ?	1

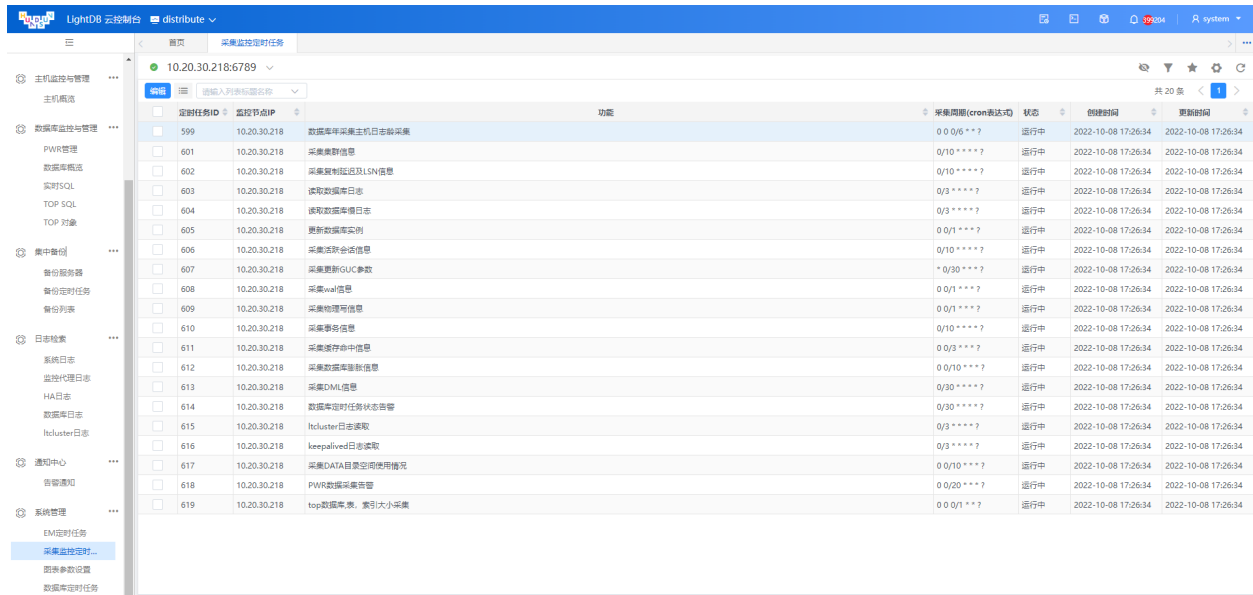


通过查看日志可以查看定时任务执行的日志明细：

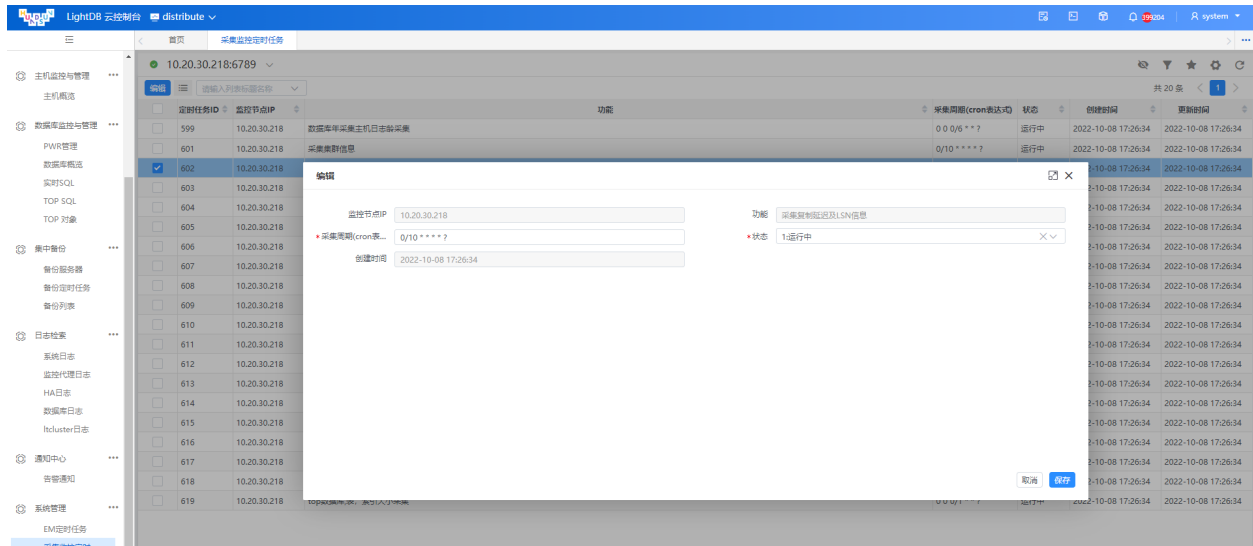


图表参数设置

图表参数配置的是 LightDB EM 每个图表的滑轨最小值和最大值，滑轨大小目前都是以毫秒为单位。



通过编辑操作，我们可以调整定时任务的采集周期和状态。采集定时任务目前使用 Spring Schedule 来实现，采集周期是标准的 cron 表达式。



采集周期 Cron 表达式是一个字符串，字符串以 5 或 6 个空格隔开，分为 6 或 7 个域，每一个域代表一个含义，Cron 有如下两种语法格式：

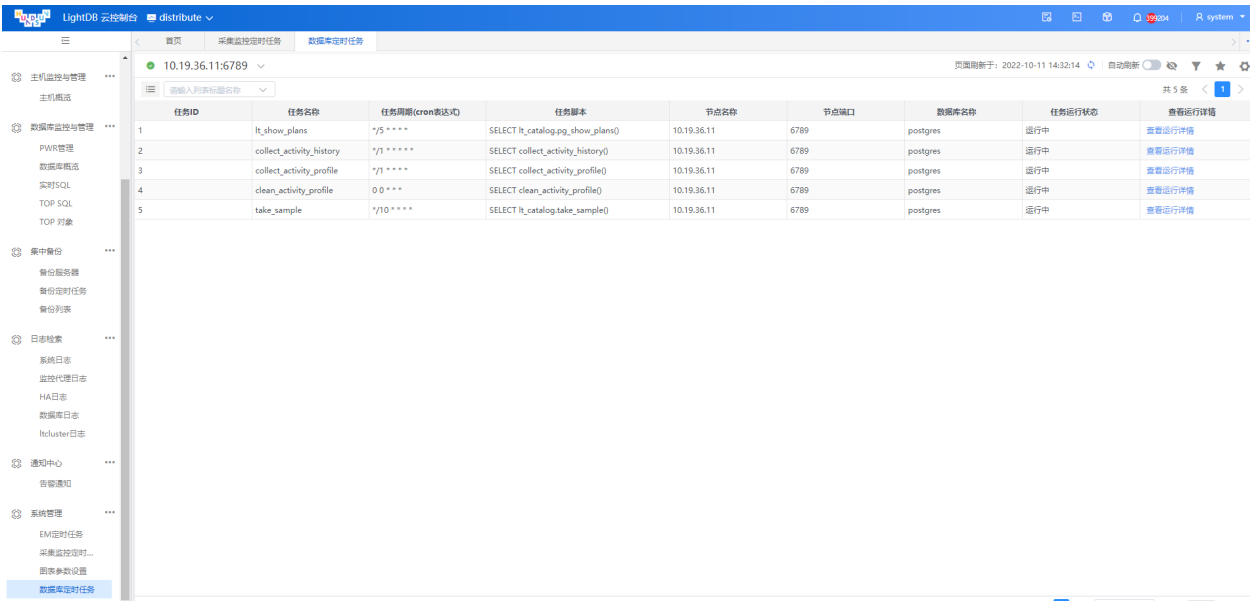
Seconds Minutes Hours DayofMonth Month DayofWeek Year 或 Seconds Minutes Hours DayofMonth Month DayofWeek

每一个域可出现的字符如下：Seconds: 可出现”，- */” 四个字符，有效范围为 0-59 的整数 Minutes: 可出现”，- */” 四个字符，有效范围为 0-59 的整数 Hours: 可出现”，- */” 四个字符，有效范围为 0-23 的整数 DayofMonth: 可出现”，- */ ? L W C” 八个字符，有效范围为 0-31 的整数 Month: 可出现”，- */” 四个字符，有效范围为 1-12 的整数或 JAN-DEc DayofWeek: 可出现”，- */ ? L C #” 四个字符，有效范围为 1-7 的整数或 SUN-SAT 两个范围。1 表示星期天，2 表示星期一，依次类推 Year: 可出现”，- */” 四个字符，有效范围为 1970-2099 年

数据库定时任务

数据库定时任务配置在表 cron.job，需要在 postgres 库中查询。目前主要四个定时任务。

定时任务	周期	功能说明
SELECT pg_show_plans()	* / 5 * * * *	清理 show plan 数据的
SELECT collect_activity_history()	* / 1 * * * * * 每秒一次	对活跃会话历史 active_session_history 和进程活动信息历史 lt_stat_activity_history 进行采样
SELECT collect_activity_profile()	* / 1 * * * * * 每分钟一次	对活跃会话汇总 active_session_profile 和进程活动汇总 lt_stat_activity_profile 进行采样
SELECT clean_activity_profile()	0 0 * * * * 每 0 点 0 分一次	汇总表清理 sample_active_session_profile、sample_activity_profile
SELECT take_sample()	* / 1 * * * * * 每分钟一次	定时生成快照



通过查看运行详情可以查看数据库定时任务执行的日志详情，方便定位问题：

帐号名称管理

帐号名	类型	昵称	皮肤	操作
2938814394@qq.com	QQ邮箱	软件174部运维	默认皮肤	设置

帐户安全

独立密码: [设置独立密码...](#)
(设置独立密码后, 进入邮箱需要输入独立密码验证, 使用QQ邮箱更加安全。)

文件夹区域加锁: [加锁“文件夹区域”...](#)
(“文件夹区域”是由“我的文件夹”, “其他邮箱”, “记事本”组成。加锁即对这几部分设置密码, 以保护你的信息。)

POP3/IMAP/SMTP/Exchange/CardDAV/CalDAV服务

开启服务:	POP3/SMTP服务 (如何使用 Foxmail 等软件收发邮件?)	已开启 关闭
	IMAP/SMTP服务 (什么是 IMAP, 它又是如何设置?)	已开启 关闭
	Exchange服务 (什么是Exchange, 它又是如何设置?)	已开启 关闭
	CardDAV/CalDAV服务 (什么是CardDAV/CalDAV, 它又是如何设置?)	已关闭 开启
	(POP3/IMAP/SMTP/CardDAV/CalDAV服务均支持SSL连接, 如何设置?)	

温馨提示: 登录第三方客户端时, 密码框需输入“授权码”进行验证①, 生成授权码

收取选项: [最近30天](#) 的邮件

- ☒ 收取“我的文件夹”
- ☒ 收取“QQ邮件订阅”
- ☒ SMTP发信后保存到服务器

(以上收取选项对POP3/IMAP/SMTP/Exchange均生效。了解更多)

☐ 收取垃圾邮件提高提醒

(该收取选项只对POP3生效。我使用了IMAP/Exchange协议, 怎么办?)

配置好钉钉以及邮箱的告警信息

LightDB 云控制台 单节点在线安装部署

10.20.31.208.5432

页面刷新于: 2022-12-23 10:34:20 自动刷新

邮件告警

邮箱地址: smtp.qq.com 邮箱用户名: 2938814394@qq.com 邮箱密码:

钉钉告警

钉钉链接: https://api.dingtalk.com/robot/send?access_token=b8b116224b4d791712bf... 钉钉密码:

[确定](#) [应用到配置](#)

配置成功后会在我们配置的钉钉群以及邮箱收到发送成功接入告警配置信息



其中邮箱接收方是在 em 的 system 修改信息处填写的邮箱地址

修改信息

* 账号

system

* 姓名

邱贤锋

外部用户名

邮箱

qiuxf41978@hundsun.com

手机

例: 15988888888

电话

例: 010-12345678

地址

杭州市滨江区江南大道3588号恒生大厦

取消

确定

配置的接受邮箱也收到成功接入告警配置的信息如下

搜索邮件和人员

收件箱

对话 (按已标记)

所有 未读 收件人是我 已标记

未标记

✓ 2938814394@qq.com

邮箱告警配置

监控目标(10.20.31.204:5434)已接入邮箱告警配置...

15:01

JIRA系统

邮箱告警配置

2938814394@qq.com

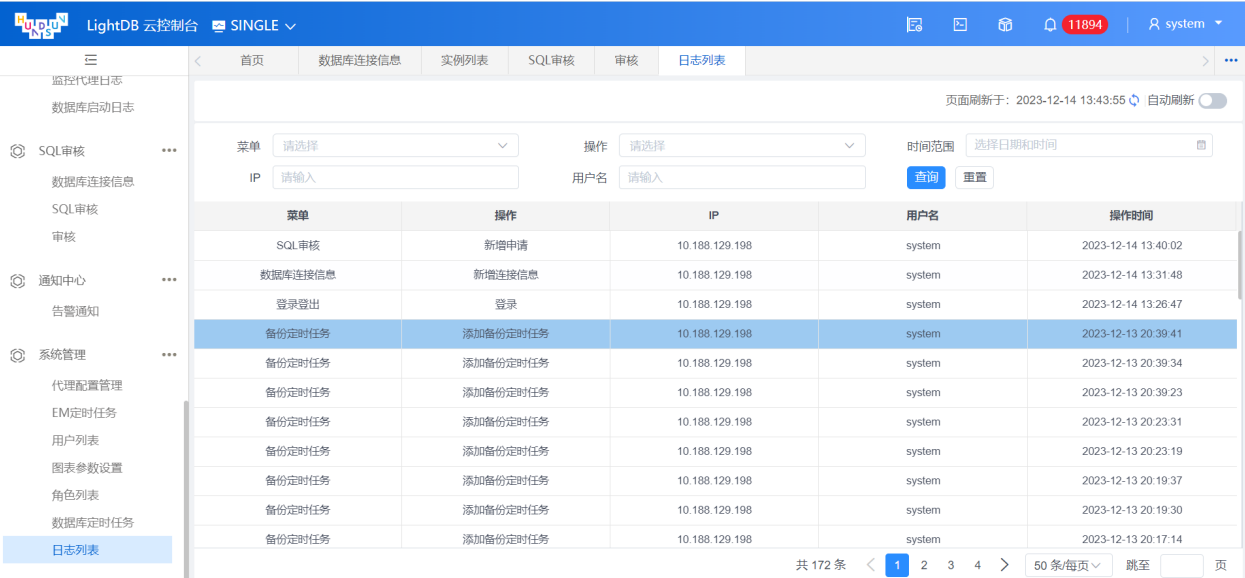
周四 2022/12/22 15:01

收件人: 邱贤锋;

监控目标(10.20.31.204:5434)已接入邮箱告警配置, 欢迎您的使用

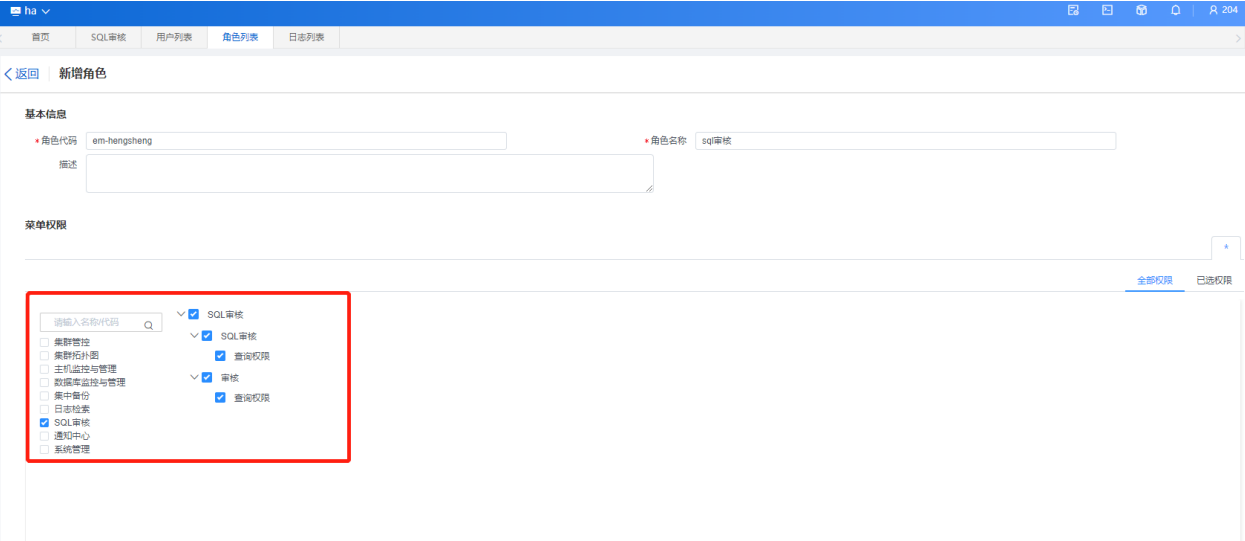
日志列表

日志列表展示的是用户的敏感操作，比如备份恢复、新增集群、修改集群、删除集群，不会记录查询操作。



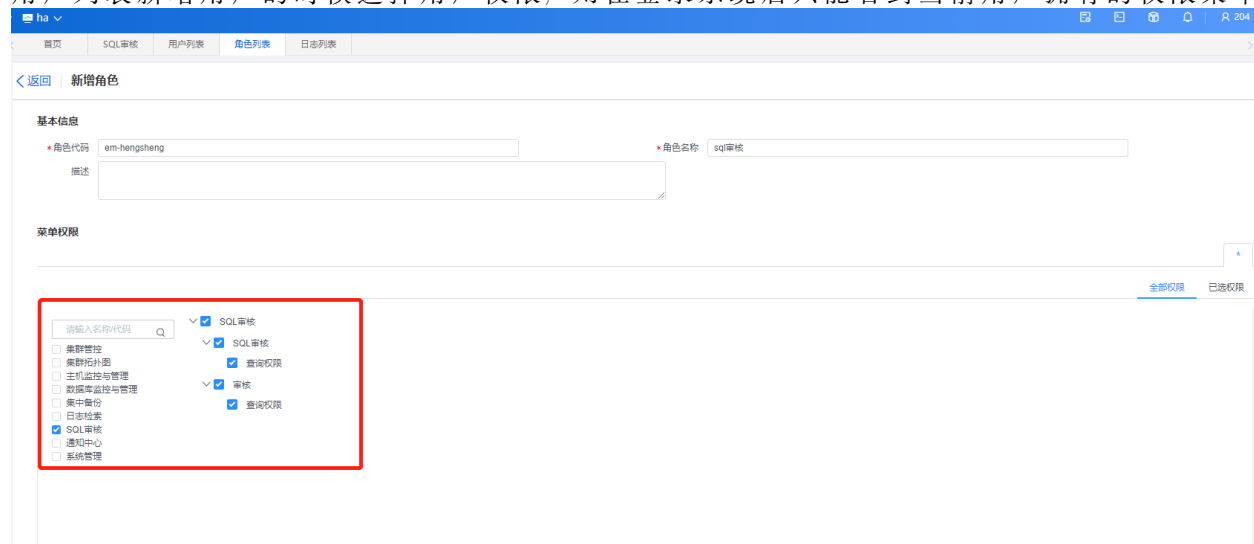
角色列表

角色列表新增角色，其中选择角色包含的菜单功能



用户列表

用户列表新增用户的时候选择用户权限，则在登录系统后只能看到当前用户拥有的权限菜单



3.15 告警待办

告警待办展示所有待处理的告警待办信息，数据库及主机信息如果超过阈值或者异常，就会自动生成告警待办。在实际运维过程中，我们需要特别关注告警待办，一般待办表示数据库及主机出现了问题，需要及时处理。



问题真正解决后，点击对应的告警待办输入确认信息保存，告警待办状态即为处理成功，并且会在顶部不再展示，只展示在已办告警列表中。

告警待办确认



任务标题 磁盘使用率告警

创建人 system

待办内容 当前主机 10.19.36.10,磁盘 /dev/loop0,磁盘使用率为 100.

00%

确认信息

请输入

取消

保存

点击查看全部进入待办列表

The screenshot shows the 'Alert Pending Confirmation' dialog box in the foreground, which is a modal window for confirming a pending alert. It contains fields for 'Task Title' (磁盘使用率告警), 'Creator' (system), 'Pending Content' (当前主机 10.19.36.10,磁盘 /dev/loop0,磁盘使用率为 100.00%), and a 'Confirmation Information' input field. At the bottom are 'Cancel' and 'Save' buttons.

In the background, the 'Alert Pending List' table is visible. It has columns for 'Alert ID', 'Alert Name', 'Alert Content', and 'Creator'. The table lists several alerts with IDs 2, all named '磁盘使用率告警', and all created by 'system'. The alert content for each entry is '当前主机 10.19.36.10,磁盘 /dev/loop0,磁盘使用率为 100.00%'. The table also shows a '一键处理' (One-click Process) button and a '处理中' (Processing) status.

告警ID	告警名称	告警内容	创建人
2	磁盘使用率告警	当前主机 10.19.36.10,磁盘 /dev/loop0,磁盘使用率为 100.00%	system
2	磁盘使用率告警	当前主机 10.19.36.10,磁盘 /dev/loop0,磁盘使用率为 100.00%	system
2	磁盘使用率告警	当前主机 10.19.36.10,磁盘 /dev/loop0,磁盘使用率为 100.00%	system
2	磁盘使用率告警	当前主机 10.19.36.10,磁盘 /dev/loop0,磁盘使用率为 100.00%	system
2	磁盘使用率告警	当前主机 10.19.36.10,磁盘 /dev/loop0,磁盘使用率为 100.00%	system

3.16 用户中心

登录用户信息栏展示当前登录用户名，鼠标移动到用户信息上会展示用户对系统信息及修改操作。



修改信息

修改信息可以修改当前用户的信息

修改信息



* 账号

system

* 姓名

外部用户名

邮箱

例: XXX@gmail.com

手机

例: 15988888888

电话

例: 010-12345678

地址

杭州市滨江区江南大道3588号恒生大厦



取消

确定

修改密码

修改密码可以修改当前用户的密码，密码强度是根据包含类型（支持大写字母、小写字母、数字、其他字符）和长度，建议使用 8 位字符串、至少包含两种字符类型；

修改密码

×

*旧密码

*新密码

*确认密码

密码强度

0%

取消

确定

锁屏

锁屏可以遮盖当前系统防止其他用户操作界面，解除锁屏需要输入当前用户的密码。

锁定登录

*

system

*

Password

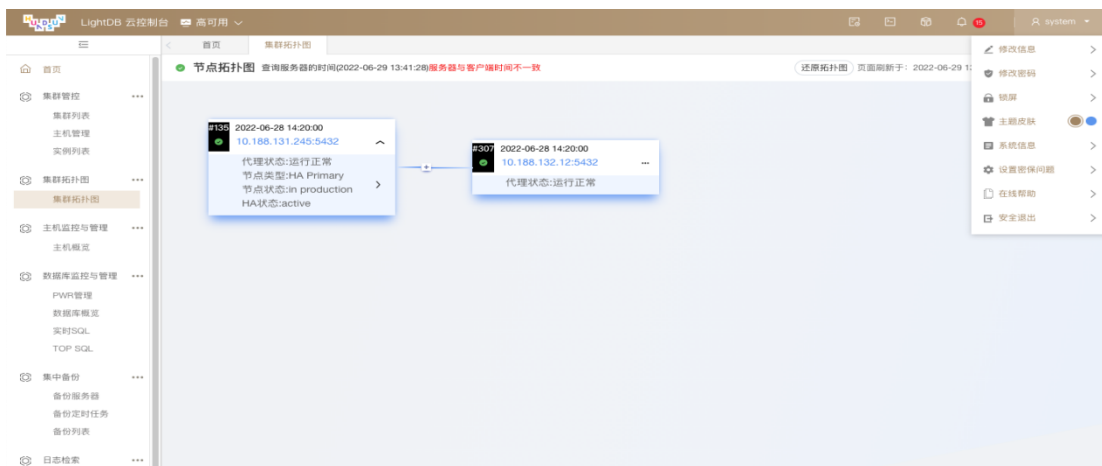
输入不能为空

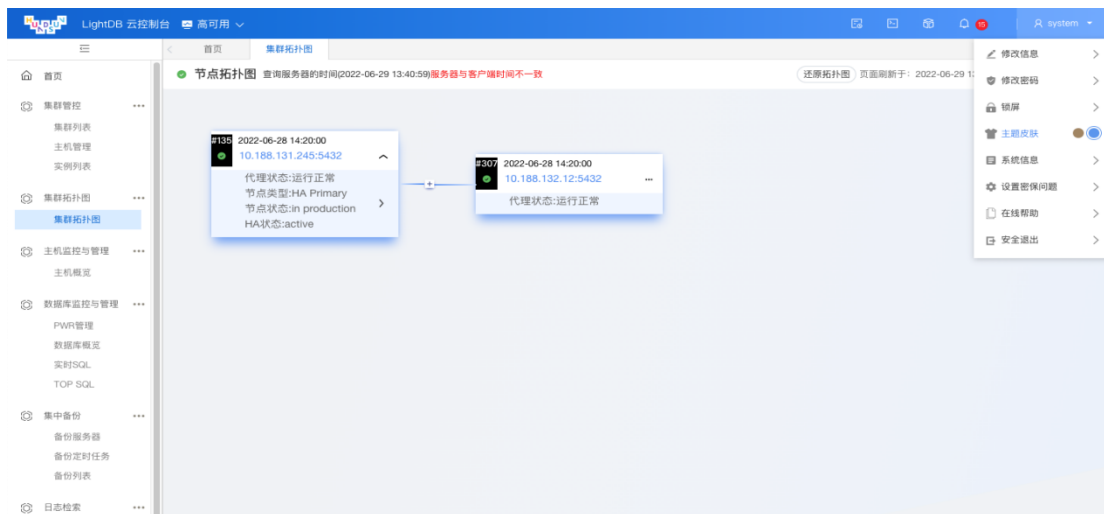
屏幕30分钟没有操作将会被锁定

登录

主题皮肤

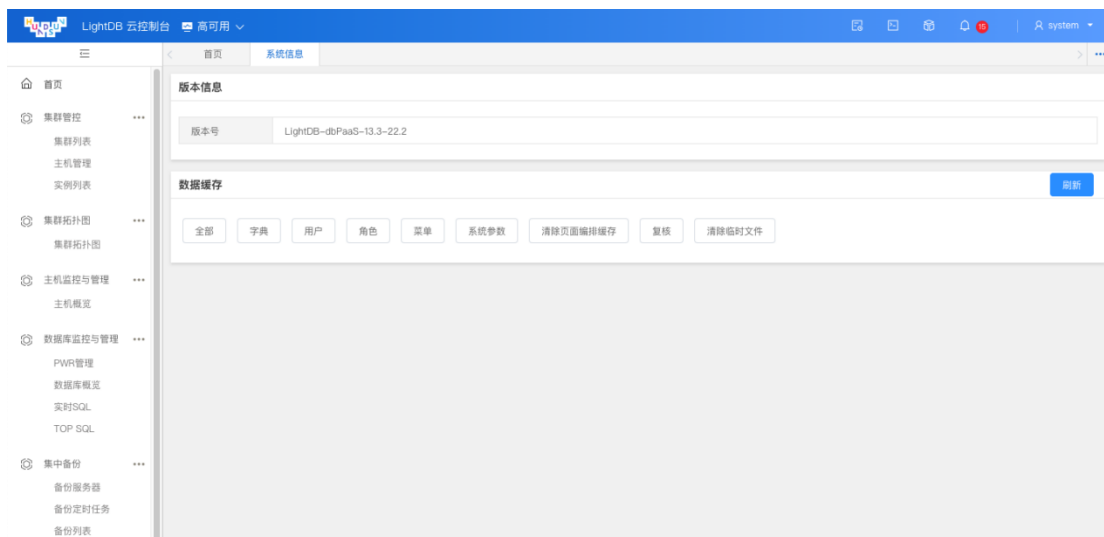
主题皮肤可以选择 2 种皮肤中的一种，咖啡色和蓝色





系统信息

系统信息目前展示的是 DBPaaS 的版本系统和缓存



设置密保问题

设置好密保问题，用户可以根据密保问题找回密码

设置密保问题

问题一

请选择

你的家乡在哪里?

你少年时代最好的朋友叫什么名字?

你上小学时最喜欢的老师姓什么?

你的理想工作是什么?

答案

问题二

请选择

你的家乡在哪里?

你少年时代最好的朋友叫什么名字?

你上小学时最喜欢的老师姓什么?

你的理想工作是什么?

答案

问题三

请选择

答案

取消

确定

在线帮助

在线帮助展示的是 DBPaaS 的用户手册



安全退出

安全退出既退出当前用户登录，进入到系统登录页面。

4 LightDB EM 注意事项

4.1 Lightdb EM 集群相关

- 1. Single（单实例）部署模式下只能添加一套主机；
- 2. HA（高可用）部署模式下，HA Primary 只能有一个；
- 3. 添加主机时需要进行测试 SSH 连接校验，测试数据库连接校验；
- 4. 代理路径目录必须为空目录且需有可执行权限；
- 5. 实例角色需要选择实际部署的角色；
- 6. 一个主机只能部署一个代理监控。

5 EM 参数

本节列出了 LightDB-EM 中系统参数的信息。

表 1: EM 参数展示

参数名	参数值说明	运维调整情况
app.name	应用名称，默认 em	无需调整
app.alias	应用别名，默认 lightdb em	无需调整

续下页

表 1 - 接上页

参数名	参数值说明	运维调整情况
app.version	应用版本号，每次发版会把版本号调整好	无需调整
app.group	应用组，每次发版会把组调整好	无需调整
server.port	应用端口，默认 17333	无需调整
ta.noWebLogUrls	不记录日志 URL	无需调整
app.web.home	应用 web 路径，一般是 http://ip:port/em	无需调整
app.web.domain	应用域名或 IP	无需调整
install.path	实际安装路径，需确保填写到安装包内部的/em 为止。例如： /data/lightdb/wuxj/LightDB1.0-em-V202302-00-000-el7-x86_64/em	无需调整
logging.config	日志配置文件路径，默认 class-path:log4j2.xml	按需调整
mybatis.mapperLocations	Mybatis mapper xml 路径	不能调整
mybatis.typeAliasesPackage	Mybatis 对象对应的包路径	不能调整
mybatis.configLocation	Mybatis 配置路径	不能调整
mybatis.basePackage	Mybatis 基础包路径	不能调整
ta.noSecurityUrls	EM 调用 AGENT 校验 URL	不能调整
ta.needSignatureUrls	EM 调用 AGENT 签名校验 URL	不能调整
dyn.spring.datasources[0].name	默认值 default，对应 em 数据库信息	不能调整
dyn.spring.datasources[0].driverClassName	数据库驱动 org.postgresql.Driver	不能调整
dyn.spring.datasources[0].url	对应 em 数据库 jdbc url 地址	按需调整
dyn.spring.datasources[0].username	对应 em 数据库用户名，默认 lightdb	按需调整
dyn.spring.datasources[0].password	对应 em 数据库密码	按需调整
dyn.spring.datasources[0].maxActive	对应 em 数据库连接池中的最大连接数，默认 208	按需调整
dyn.spring.datasources[0].maxWait	从 em 数据库连接池中获取连接的最大等待时间，单位 ms，运维可以根据实际调整优化	按需调整
dyn.spring.datasources[0].maxIdle	对应 em 数据库最大空闲连接数，默认 1	按需调整
dyn.spring.datasources[0].removeAbandoned	超过时间限制是否回收，默认 true	按需调整
dyn.spring.datasources[0].removeAbandonedT	超过时间限制多长，默认 600s，	按需调整
dyn.spring.datasources[0].logAbandoned	指定发生 removeabandoned 的时候，是否记录当前线程的堆栈信息到日志中，默认 true	按需调整
spring.redis.database	redis 库序号，默认 0	无需调整
spring.redis.host	redis 主机 ip	无需调整
spring.redis.port	redis 端口 port	无需调整
spring.redis.password	redis 密码	无需调整
spring.redis.timeout	redis 超时时间，默认 60000	无需调整
notify.mail.host	邮件告警的邮件服务器	按需调整
notify.mail.account	邮件告警的发送邮箱账号	按需调整
notify.mail.password	邮件告警的发送邮箱密码	按需调整
notify.dingding.baseurl	钉钉告警的机器人地址如： https://oapi.dingtalk.com/robot/send?access_token=3b9eb5e97ec0ab32c39a1c0709321440cf2115	按需调整
notify.dingding.secret	钉钉告警的机器人的加签秘钥，如：SEC5720e84b2c0a2694255237b99375a2a836	按需调整

1896d95

,3431a56df

6 AGENT 参数

本节列出了 LightDB-EM 中 AGENT 代理系统参数相关信息。

表 2: AGENT 代理参数展示

参数名	参数值说明	运维调整情况
app.name	应用名称，默认 em	无需调整
app.alias	应用别名，默认 lightdb em	无需调整
app.version	应用版本号，每次发版会把版本号调整好	无需调整
app.group	应用组，每次发版会把组调整好	无需调整
server.port	应用端口，默认 17333	无需调整
ta.noWebLogUrls	不记录日志 URL	无需调整
app.web.home	应用 web 路径，一般是 http://ip:port/em	无需调整
app.web.domain	应用域名或 IP	无需调整
install.path	实际安装路径，需确保填写到安装包内部的/em 为止。例如： /data/lightdb/wuxj/LightDB1.0-em-V202302-00-000-el7-x86_64/em	无需调整
logging.config	日志配置文件路径，默认 class-path:log4j2.xml	按需调整
mybatis.mapperLocations	Mybatis mapper xml 路径	不能调整
mybatis.typeAliasesPackage	Mybatis 对象对应的包路径	不能调整
mybatis.configLocation	Mybatis 配置路径	不能调整
mybatis.basePackage	Mybatis 基础包路径	不能调整
ta.noSecurityUrls	EM 调用 AGENT 校验 URL	不能调整
ta.needSignatureUrls	EM 调用 AGENT 签名校验 URL	不能调整
dyn.spring.datasources[0].name	默认值 default，对应 em 数据库信息	不能调整
dyn.spring.datasources[0].driverClassName	数据库驱动 org.postgresql.Driver	不能调整
dyn.spring.datasources[0].url	对应 em 数据库 jdbc url 地址	按需调整
dyn.spring.datasources[0].username	对应 em 数据库用户名，默认 lightdb	按需调整
dyn.spring.datasources[0].password	对应 em 数据库密码	按需调整
dyn.spring.datasources[0].maxActive	对应 em 数据库连接池中的最大连接数，默认 208	按需调整
dyn.spring.datasources[0].maxWait	从 em 数据库连接池中获取连接的最大等待时间，单位 ms，运维可以根据实际调整优化	按需调整
dyn.spring.datasources[0].maxIdle	对应 em 数据库最大空闲连接数，默认 1	按需调整
dyn.spring.datasources[0].removeAbandoned	超过时间限制是否回收，默认 true	按需调整
dyn.spring.datasources[0].removeAbandonedT	超过时间限制多长，默认 600s，	按需调整
dyn.spring.datasources[0].logAbandoned	指定发生 removeabandoned 的时候，是否记录当前线程的堆栈信息到日志中，默认 true	按需调整
dyn.spring.datasources[1].name	对应被监控数据库实例信息。默认格式为 IP:PORT	按需调整
dyn.spring.datasources[1].driverClassName	数据库驱动 org.postgresql.Driver	按需调整
dyn.spring.datasources[1].url	对应被监控数据库实例 jdbc url 地址	按需调整
dyn.spring.datasources[1].username	对应被监控数据库实例用户名，默认 lightdb	按需调整
dyn.spring.datasources[1].password	对应被监控数据库实例密码	按需调整

续下页

表 2 - 接上页

参数名	参数值说明	运维调整情况
dyn.spring.datasources[1].maxActive	对应被监控数据库实例连接池中的最大连接数，默认 208	按需调整
dyn.spring.datasources[1].maxWait	被监控数据库实例连接池中获取连接的最大等待时间，单位 ms，运维可以根据实际调整优化	按需调整
dyn.spring.datasources[1].maxIdle	被监控数据库实例最大空闲连接数，默认 1	按需调整
dyn.spring.datasources[1].removeAbandoned	超过时间限制是否回收，默认 true	按需调整
dyn.spring.datasources[1].removeAbandonedTimeout	超过时间限制多长，默认 600s，	按需调整
dyn.spring.datasources[1].logAbandoned	指定发生 removeAbandoned 的时候，是否记录当前线程的堆栈信息到日志中，默认 true	按需调整
host.ip	被监控主机 IP	不能调整
host.hostname	被监控主机名	不能调整
host.user	被监控主机用户名	不能调整
database.server.url	被监控数据库 URL。默认为 IP:PORT 格式	不能调整
database.server.username	被监控数据库名称	不能调整
database.server.type	数据库类型 1:lightdb 2:postgresql	不能调整
database.server.cluster.mode	数据库部署类型，0 单实例 1 高可用 2 分布式	不能调整
database.server.cluster.nodetype	数据库节点类型，Single Only、HA Primary、HA Standby、HA Witness、Distribute Coordinator Primary、Distribute Coordinator Standby、Distribute Coordinator Witness、Distribute Worker Primary、Distribute Worker Standby、Distribute Worker Witness	不能调整
lightdb.agent.monitorType	代理监控类型，1：监控主机 2：监控实例 3：em 定时任务	不能调整
spring.redis.database	redis 库序号，默认 0	按需调整
spring.redis.host	redis 主机 ip。如果从安装包拷贝部署，需要替换 \${em_host} 为当前服务器 IP	按需调整
spring.redis.port	redis 端口 port。如果从安装包拷贝部署，需要替换 \${redis_port} 为 redis 提供服务的端口	按需调整
spring.redis.password	redis 密码。如果从安装包拷贝部署，需要替换 \${redis_password} 为 redis 配置的密码	按需调整
spring.redis.timeout	redis 超时时间，默认 60000	按需调整
notify.mail.host	邮件告警的邮件服务器	按需调整
notify.mail.account	邮件告警的发送邮箱账号	按需调整
notify.mail.password	邮件告警的发送邮箱密码	按需调整
notify.dingding.baseurl	钉钉告警的机器人地址如： https://oapi.dingtalk.com/robot/send?access_token=3b9eb5e97ec0ab32c39a1c0709321440cf2115	按需调整
notify.dingding.secret	钉钉告警的机器人的加签秘钥，如：SEC5720e84b2c0a2694255237b99375a2a836	按需调整

续下页

表 2 - 接上页

参数名	参数值说明	运维调整情况
lightdb.jieba.noneStopChars	非停止符，默认为:,-,_,= 多个符号使用 ‘,’ 分割用于在分词时将由非停止符连接的多个单次拼接成一个单词进行搜索，必须在 agent 和 em 同时配置。	按需调整

7 EM 错误码

本节列出了 LightDB-EM 中错误码相关信息。

表 3: EM 错误码展示

错误码	中文描述	原因	解决方法
LEG01	请求参数校验异常	通用参数校验不通过	传入正确的请求参数
LEG02	请求参数为空	请求参数为空	传入正确的请求参数
LEG03	文件上传失败	文件大于默认上传大小 1024M	修改 jrescloud.properties 配置文件中 spring.http.multipart.maxFileSize=1024MB spring.http.multipart.maxRequestSize=1024MB spring.servlet.multipart.max-file-size=1024MB spring.servlet.multipart.max-request-size=1024MB
LEG04	文件下载失败	没有权限	为文件存储目录赋予 lightdb 用户权限以及可读写权限
LEG05	文件不存在	文件不存在	确保文件真实存在
LEG06	目录不存在	目录不存在	创建目录
LEG07	创建目录失败	没有权限/磁盘空间不足	为文件存储目录赋予 lightdb 用户权限以及可读写权限
LEG08	文件关闭失败	文件不存在或无法访问/文件已被其他进程打开/操作系统资源限制	确保文件路径存在、能正常访问且有权限/释放文件锁或资源
LEG09	文件打开失败	文件打开失败	确保文件路径存在、能正常访问且有权限/释放文件锁或资源
LEG10	文件写入失败	文件写入失败	确保文件路径存在、能正常访问且有权限/释放文件锁或资源
LEG11	内部服务调用失败 (em 调用 agent 接口)	agent 异常或已停止	检查 agent 状态重启 agent 代理
LEG12	无操作权限，请联系管理员授权	用户未赋权	用管理员账号赋予用户菜单等权限
LEG13	HTTP 请求失败 (外部服务 web 调用 em 接口)	未授予访问权限	开放鉴权秘钥
LEG14	服务器时间不同步	服务器之间时间不同步，存在偏差	手动调整服务器时间/使用 NTP (Network Time Protocol)
LEG15	agent 故障，请检查 agent 的状态	agent 异常或已停止	检查 agent 状态重启 agent 代理

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEG16	查询失败,统一处理查询操作	数据库连接异常/网络故障导致与数据库之间的连接断开	检查数据库状态网络状态内存资源等
LEG17	修改失败,统一处理更新操作	数据库连接异常/网络故障导致与数据库之间的连接断开	检查数据库状态网络状态内存资源等
LEG18	删除失败,统一处理删除操作	数据库连接异常/网络故障导致与数据库之间的连接断开	检查数据库状态网络状态内存资源等
LEG19	添加失败,统一处理添加操作	数据库连接异常/网络故障导致与数据库之间的连接断开	检查数据库状态网络状态内存资源等
LEG20	字段排序不支持,统一处理添加操作	该字段不支持排序	请选择其他字段进行排序
LEG21	解析失败!,统一处理解析操作	转码解码异常	检查编码格式和字符集 utf-8
LEG22	将文件 scp 到远程服务器报错	目录不存在/没有权限/磁盘空间不足	确保目录赋予 lightdb 用户权限以及可读写权限/磁盘空间充足
LEG23	在远程服务器执行命令出错	没有权限/没有那个文件或目录	确保目录赋予 lightdb 用户权限以及可读写权限/确保文件存在且有权限执行
LEG24	数据库表记录已存在	主键/唯一键冲突	确保数据是否已经添加,无需重复添加/更换内容再次添加
LEG25	数据字典编码 dictCode 不能为空	数据字典 code 为空	确保传入的字典 key 为非空
LEG26	shell 脚本在 em 本地执行失败	创建脚本 lt_probackup_for_em.sh, 并将 lt_probackup 子命令 archive-push 附加到脚本中异常/创建脚本 lt_probackup_for_em.sh 并清空脚本异常	确保目录赋予 lightdb 用户权限以及可读写权限/确保文件存在且有权限执行
LEG27	执行 shell 脚本的所需参数个数不对	执行 shell 脚本的所需参数个数不对	传入正确的请求参数
LEG28	hostIp 为空	hostIp 为空	传入正确的 ip
LEG29	hostIp 不合法	hostIp 不合法	传入正确的 ip
LEG30	contactType 为空	contactType 为空	传入正确的联系人类型
LEG31	contactType 内容不合法	contactType 内容不合法	传入正确的联系人类型
LEG32	order 字段内容只能为 asc 或者 desc	order 字段内容只能为 asc 或者 desc	order 字段内容只能为 asc 或者 desc
LEG33	该 sort 字段不支持排序	该 sort 字段不支持排序	请选择其他字段进行排序
LEG34	未给操作用户配置 sudo 免密	未给 lightdb 操作用户配置 sudo 免密	给 lightdb 用户配置 sudo 免密
LEG35	上传的文件类型不支持,请上传 zip 文件	上传的文件类型不支持,请上传 zip 文件	上传的文件类型不支持,请上传 zip 文件
LEG36	配置项: 单个文件大小配置得不准确, 格式为 xxxMB	配置项: 单个文件大小配置得不准确, 格式为 xxxMB	修改配置项: 格式为 xxxMB

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEG37	上传的文件过大	文件大于默认上传大小 1024M	修改 jrescloud.properties 配置文件中 spring.http.multipart.maxFileSize=1024MB spring.http.multipart.maxRequestSize=1024MB spring.servlet.multipart.max-file-size=1024MB spring.servlet.multipart.max-request-size=1024MB
LEG38	上传的文件总大小过大	文件大于默认上传大小 1024M	修改 jrescloud.properties 配置文件中 spring.http.multipart.maxFileSize=1024MB spring.http.multipart.maxRequestSize=1024MB spring.servlet.multipart.max-file-size=1024MB spring.servlet.multipart.max-request-size=1024MB
LEG39	上传的文件名称不合法	上传的文件名称不合法	NO USE
LEG40	agentId 不能为空	agentId 不能为空	确认代理 id 不为空
LEG41	agent 无效	代理信息不存在	确认正确添加代理
LEG42	agent 路径不能为空	agent 路径不能为空	请传入正确的绝对路径
LEG43	agent 路径不合法	agent 路径不合法	请传入正确的绝对路径
LEG44	agent 端口不能为空	agent 端口不能为空	请传入正确的代理端口 (1024,65535]
LEG45	agent 压缩包路径不能为空	agent 压缩包路径不能为空	请确认传入正确的 agent 压缩包路径
LEG46	agent 压缩包路径不合法	agent 压缩包路径不合法	请传入正确的绝对路径
LEG47	agent_stop 脚本在主机上不存在!	agent_stop 脚本在主机上不存在!	请确认是否删除了代理安装路径下的系统文件, 如果被人为删除, 请手动停止代理, 删除 em 库下 lem_instance、lem_host、lem_agent 相关数据, 重新安装代理。
LEG48	文件不存在	文件不存在	请确认服务器上文件是否存在, 如果被人为删除, 请手动停止代理, 删除 em 库下 lem_instance、lem_host、lem_agent 相关数据, 重新安装代理
LEG49	agent 状态为空, 不再执行后续动作	agent 状态为空, 不再执行后续停止实例代理, 删除服务器上代理目录和删除代理表记录动作	确认代理状态是否为空, 如为空请重新添加代理
LEG50	服务器空闲内存不足	服务器空闲内存不足	确保服务器内存大于 1G
LEG51	只支持 22.3 及以上版本	只支持 22.3 及以上版本数据库	如要使用该功能请升级 lightdb 数据库大于 22.3 版本
LEG52	实例巡检报错	没有权限/没有 python3 环境	请检查代理路径下所有文件的权限/安装 python3 环境

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEG53	读取实例检测文件失败	读取实例检测文件失败/没有权限/文件不存在	请检查代理路径下所有文件的权限/确保实例检测文件没有被删除 lightdb_maa_checklist_xxx.html
LEG54	instanceIp 为空	instanceIp 为空	请传入正确的实例 IP
LEG55	instancePort 为空	instancePort 为空	请传入正确的实例端口
LEG56	pwr 报告生成, 不支持实例角色 witness	pwr 报告生成, 不支持实例角色 witness	请用主备节点生成 pwr 报告
LEG57	实例巡检报告名称为空	实例巡检报告名称为空	请传入正确的实例巡检报告名称/xxx/xxx/lightdb_maa_checklist_xxx.html
LEG58	获取实例巡检报告列表失败	获取实例巡检报告列表失败	确保该命令能正常执行 cd agentHomePath/check_list;ls lightdb_maa_checklist*.html -lr head -n 100 awk '{printf "%sn", \$9;}'
LEG59	当前 lightdb 数据库版本不支持	当前 lightdb 数据库版本不支持	如要使用该功能请升级 lightdb 数据库大于 22.4 版本
LEG60	集群下数据库实例版本为空	集群下数据库实例版本为空	请确认集群下添加的数据库版本一致
LEG61	判断 python 环境失败	判断 python 环境失败	确保该命令能正常执行 python -V 2>&1
LEG62	需要 python2 环境	需要 python2 环境	需要 python2 环境
LEG63	需要 python3 环境	需要 python3 环境	需要 python3 环境
LEG64	根据 cn 节点信息获取 dn 列表失败	根据 cn 节点信息获取 dn 列表失败	确保数据库集群正常
LEG65	分布式集群备份不支持单机多实例	分布式集群备份不支持单机多实例	分布式集群备份不支持单机多实例, 如要使用备份功能请升级为多服务器分布式模式
LEG66	危险路径, 不允许删除	危险路径, 不允许删除	确保路径是绝对路径没有 * 等
LEG67	当前实例不是原生 PG	当前实例不是原生 PG	请安装原生 PG
LEG68	原生 PG 不支持数据库定时任务	原生 PG 不支持数据库定时任务	请使用 lightdb 数据库
LEG69	原生 PG 未安装 lightdb 提供的插件	原生 PG 未安装 lightdb 提供的插件	NO USE
LEG70	原生 PG 插件冲突	原生 PG 插件冲突	NO USE
LEG71	原生 PG 安装了部分 lightdb 插件	原生 PG 安装了部分 lightdb 插件	NO USE
LEG72	执行远程命令判断实例是否是原生 PG 失败	执行远程命令判断实例是否是原生 PG 失败	NO USE
LEG73	执行远程命令判断数据库类型失败	执行远程命令判断数据库类型失败	NO USE
LEG74	检查原生 PG 安装 lightdb 插件情况	检查原生 PG 安装 lightdb 插件情况	NO USE
LEG75	实例巡检文件名不合法	实例巡检文件名不合法	文件名需满足正则 ^lightdb_maa_checklist_*.html\$
LEG76	上传的 agent 文件名不合法	上传的 agent 文件名不合法	文件名需满足正则 ^(LightDBd.d-agent-V.*lightdb-agent-.*).zip\$
LEG99	其他异常	系统异常	请联系运维人员处理

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEC01	集群查询失败!	数据库连接异常/网络故障导致与数据库之间的连接断开	检查数据库状态网络状态内存资源等
LEC02	集群修改失败!	数据库连接异常/网络故障导致与数据库之间的连接断开	检查数据库状态网络状态内存资源等
LEC03	集群删除失败!	数据库连接异常/网络故障导致与数据库之间的连接断开	检查数据库状态网络状态内存资源等
LEC04	集群添加失败!	数据库连接异常/网络故障导致与数据库之间的连接断开	检查数据库状态网络状态内存资源等
LEC05	集群不存在!	集群不存在!	该集群是否已被删除, 已删除则忽略
LEC06	集群已存在!	集群已存在!	唯一键冲突, 请重新命名集群名称
LEC07	集群已关联实例, 请先删除实例再删集群!	集群已关联实例, 请先删除实例再删集群!	删除集群前先删除集群下的实例信息
LEC08	集群已关联实例, 无法修改部署模式与数据库类型!	集群已关联实例, 无法修改部署模式与数据库类型!	如要修改请重新部署
LEC09	集群 id 不能为空	集群 id 不能为空	请传入正确的集群 id
LEC10	集群名称已存在	集群名称已存在	集群名称必须唯一, 请重新命名
LEC11	集群的数据库版本为空	集群的数据库版本为空	NO USE
LEC12	集群高可用部署, 节点的数据库版本不一致	集群高可用部署, 节点的数据库版本不一致	确保各个节点上部署的数据库是相同的版本
LEC13	集群高可用部署, 节点的 LTHOME 或者 LTDATA 不一致	节点的 LTHOME 或者 LTDATA 不一致	确保各个节点上对应的 LTHOME 或者 LTDATA 是相同的
LEC14	集群高可用部署, 集群节点的 LTDATA 目录不空	集群节点的 LTDATA 目录不空	集群高可用部署之前确保各个节点的 LTDATA 是空的
LEC15	集群高可用部署, 不符合一主(primary) 至少一从(standby), witness 可选的原则	不符合一主(primary) 至少一从(standby), witness 可选的原则	注意各个节点角色分配, 要符合一主(primary) 至少一从(standby), witness 可选的原则
LEC20	检查 keepalived 和 ltclusterd 进程是否存在出错	keepalived 和 ltclusterd 进程是否存在检查失败	确认各节点 keepalived 和 ltclusterd 进程存在且状态是正常的
LEC21	新增操作系统用户 lightdb 的 bash 环境变量出错	新增操作系统用户 lightdb 的 bash 环境变量出错	EM 通过 ssh 协议连接到集群节点, 请确认当前用户有操作权限
LEC22	初始化数据库出错	初始化数据库出错	EM 通过 ssh 协议连接到集群节点, 请确认当前用户有操作权限, 配置的数据库初始化参数正确
LEC23	修改 \$LTDATA/lightdb.conf 文件中配置项出错	修改 \$LTDATA/lightdb.conf 文件中配置项出错	确认配置文件存在, 当前用户有读写权限
LEC24	启动数据库实例出错	启动数据库实例出错	确认 ltbase/script/ha/6_lightdb_start.sh 脚本存在, 且当前用户有执行权限
LEC25	验证数据库实例启动出错	验证数据库实例启动出错	确认 ltbase/script/ha/7_lightdb-validate-start.sh 脚本存在, 且当前用户有执行权限

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEC26	初始化一个 ltcluster 安装并注册集群主节点, 手动启动一个 ltclusterd 守护进程。出错	初始化一个 ltcluster 安装并注册集群主节点, 手动启动一个 ltclusterd 守护进程。出错	确认脚本 ltatse/script/ha/9.1_start_primary_ltclusterd.sh 存在且当前用户有操作权限, 可尝试手动执行, 确认异常点
LEC27	遍历集群从节点, 从该 LightDB 主节点克隆一个 LightDB 备份节点出错	从 LightDB 主节点克隆一个 LightDB 备份节点出错	确认脚本 ltatse/script/ha/9.2_start_standby_ltcluster.sh 存在且当前用户有操作权限, 可尝试手动执行, 确认异常点
LEC28	高可用集群主节点 (实例角色 primary) 重新启动。出错	高可用集群主节点 (实例角色 primary) 重新启动。出错	请确认脚本 ltatse/script/ha/9.3_restart_primary.sh 存在且当前用户有操作权限, 可尝试手动执行, 确认异常点
LEC29	在集群从节点 (实例角色 standby) 启动 ltcluster 服务。出错	在集群从节点 (实例角色 standby) 启动 ltcluster 服务。出错	请确认脚本 ltatse/script/ha/9.4_start_standby_ltclusterd.sh 存在且当前用户有操作权限, 可尝试手动执行, 确认异常点
LEC30	启动集群 witness 节点出错	启动集群 witness 节点出错	请确认脚本 ltatse/script/ha/9.5_start_witness.sh 存在且当前用户有操作权限, 可尝试手动执行, 确认异常点
LEC31	在集群非 witness 节点安装和配置 keepalived 出错	在集群非 witness 节点安装和配置 keepalived 出错	请确认脚本 ltatse/script/ha/10_generate_keepalived_conf.sh 存在且当前用户有操作权限, 可尝试手动执行, 确认异常点
LEC32	遍历高可用集群从节点, em 执行 shell 脚本, 数据库 base 目录/ha/12_generate_sql_detector.sl 启动探针失败	em 执行脚本失败	请确认脚本 ltatse/script/ha/12_generate_sql_detector.sh 存在且当前用户有操作权限, 可尝试手动执行, 确认异常点
LEC33	em 在集群节点检测 ntp 服务失败	em 在集群节点检测 ntp 服务失败	确认脚本 ltatse/script/ha/10.2_ntp_check.sh 脚本存在, 且当前用户有相应执行权限, 可尝试手动执行, 确认异常点
LEC34	集群节点部分有 ntp 服务, 部分没有 ntp 服务	集群节点部分有 ntp 服务, 部分没有 ntp 服务	确认集群各个节点 ntp 服务正常
LEC35	在集群节点生成 hba 文件失败	在集群节点生成 hba 文件失败	确认脚本 ltatse/script/ha/0.0.3_generate_pg_hba.sh 脚本存在, 且当前用户有相应执行权限, 可尝试手动执行, 确认异常点
LEC36	创建用于数据库实例备份的数据库用户 lt_probackup 出错	创建用于数据库实例备份的数据库用户 lt_probackup 出错	确认脚本 ltatse/script/ha/8.1_create_lt_probackup.sh 脚本存在, 且当前用户有相应执行权限, 可尝试手动执行, 确认异常点
LEC37	集群节点角色 (role) 不能为空	集群节点角色 (role) 不能为空	确认为集群中的节点选定了角色
LEC38	节点角色 (role) 错误, role 范围 (primary,standby,witness)	节点角色 (role) 错误, role 范围 (primary,standby,witness)	确认集群中的节点角色是 (primary,standby,witness) 中的一个

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEC39	priority 不能为空	priority 不能为空	确认为集群中的节点设定了优先级
LEC40	priority 错误, 范围 0 或者 100	priority 错误, 范围 0 或者 100	确认优先级设定的是 0 或者 100
LEC41	sync 字段不能为空	sync 字段不能为空	确认勾选是否同步选项
LEC42	sync 字段错误, sync 取值范围 [sync,async,none]	sync 取值范围 [sync,async,none]	是否同步选项范围为 [sync,async,none]
LEC43	集群高可用部署, 节点的 ip 需要均不同	集群高可用部署, 节点的 ip 需要均不同	确认集群中节点 IP 没有重复的
LEC44	在集群节点生成 ntp.conf 文件失败	在集群节点生成 ntp.conf 文件失败	确认当前用户有对应节点上的读写权限
LEC45	实例启停, 不支持原生 PG	实例启停, 不支持原生 PG	不支持原生 PG 的在线启停, 可以根据需要手动启停
LEC46	实例卸载, 不支持原生 PG	实例卸载, 不支持原生 PG	不支持原生 PG 的在线卸载, 可以根据需要手动卸载
LEC47	新建集群, 不支持原生 PG	新建集群, 不支持原生 PG	不支持原生 PG 集群的在线部署
LEC48	数据库实例所属集群类型不支持	数据库实例所属集群类型不支持	NO USE
LEH01	暂无主机	暂无主机	新增一个主机
LEH02	主机故障	主机故障	请检查主机是否存在异常
LEH03	主机查询失败!	主机查询失败!	主机信息查询失败, 确认 EM 连接的数据库正常
LEH04	主机修改失败!	主机修改失败!	NO USE
LEH05	主机删除失败!	删除主机 agent 记录, 删除主机记录, 删除主机采集定时任务失败	确认 EM 连接的数据库正常
LEH06	主机添加失败!	主机添加失败!	EM 通过 ssh 协议连接到主机, 请确认所操作主机可达, 且有相应的操作权限
LEH07	主机不存在!	主机不存在!	确认想要操作的主机是否正常添加
LEH08	主机已存在!	主机已存在!	不要重复添加同一个主机
LEH09	主机连接失败!	主机连接失败!	EM 通过 ssh 协议连接到主机, 请确认所操作主机可达
LEH10	主机密码解密失败!	主机密码解密失败!	EM 通过 ssh 协议连接到主机, 请确认所操作主机密码是正确的
LEH11	agent 部署失败!	agent 部署失败!	主机代理部署异常, 请尝试重新添加
LEH12	主机代理已存在!	主机代理已存在!	不要重复添加相同的主机代理
LEH13	agent 上传失败!	agent 上传失败!	代理包上传失败, 确认服务器可达, 上传路径正确且有相应权限
LEH14	端口已被占用!	端口已被占用!	切换一个未被占用的端口号
LEH15	agent 路径无权限!	agent 路径无权限!	确认上传路径有写入读写权限
LEH16	agent 停止失败!	agent 停止失败!	确认代理进程存活正常, 或者手动停止异常代理进程

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEH17	agent 校验脚本上传失败!	agent 校验脚本上传失败!	确认服务器可达且有执行权限
LEH18	agent 路径校验失败!	agent 路径校验失败!	确认代理路径存在
LEH19	agent 路径下存在文件或目录, 请更换路径!	agent 路径下存在文件或目录, 请更换路径!	确认主机代理存放路径下是空的
LEH20	agent 线程已存在, 请先停止旧线程!	agent 线程已存在, 请先停止旧线程!	主机代理进程已经存在, 可以先停止旧进程, 再重新添加
LEH21	主机、代理、实例信息无变化!	主机、代理、实例信息无变化!	确认主机、代理、实例信息至少有一个变化
LEH22	single, 模式只能添加一个主机!	single, 模式只能添加一个主机!	确认 single 模式下只添加了一个主机
LEH23	高可用模式只能添加一个主节点!	高可用模式只能添加一个主节点!	确认高可用模式部署下, 只添加了一个主节点 (primary)
LEH24	agent 端口为空	agent 端口为空	确认代理端口不要为空
LEH25	主机或代理异常	主机或代理异常	确认主机或者代理状态是否正常
LEH26	获取主机检验信息失败!	获取主机检验信息失败!	确认 agent-Path/scripts/check_host_param.sh 存在且有相应的执行权限
LEH27	该主机上存在实例信息, 请先删除实例!	该主机上存在实例信息	确认删除旧实例之后再添加新的实例
LEH28	主机上有数据库安装目录, 请先删除数据库安装目录	数据库安装目录为空	确认先删除数据库安装目录
LEH29	主机和主机代理信息无变化!	主机和主机代理信息无变化!	确认修改时主机和代理信息要和原有信息不同
LEH30	hostId 不能为空	hostId 不能为空	确认主机 ID 不能为空
LEH31	hostIp 不能为空	hostIp 不能为空	确认主机 IP 不能为空
LEH32	hostIp 不合法	hostIp 不合法	确认主机 IP 要合法
LEH33	ssh 端口不能为空	ssh 端口为空	确认 SSH 端口不为空
LEH34	主机用户不能为空	主机用户为空	确认主机用户不能为空
LEH35	主机用户密码不能为空	主机用户密码为空	确认主机用户密码不能为空
LEH36	主机 id 与主机 ip 不匹配	主机 id 与主机 ip 不匹配	确认操作的主机 ID 和主机 IP 相匹配
LEH37	主机开机时间不能为空	主机开机时间为空	确认主机开机时间不能为空
LEH38	主机记录创建时间不能为空	主机记录创建时间为空	确认主机记录创建时间不能为空
LEH39	agent 启动失败!	agent 启动失败!	agent 启动失败!, 可根据启动失败的提示确认原因
LEH40	删除实例代理和实例采集定时任务!	删除实例代理和实例采集定时任务失败	确认 EM 系统运行正常, 可稍后重试
LED00	采集记录不存在	采集记录不存在	确认 EM 系统运行正常且所查询的主机正常被监控
LED01	数据库实例不存在	数据库实例不存在	确认 EM 系统运行正常且所查询的数据库实例正常被监控,
LED02	数据库不存在		NO USE
LED03	数据库表不存在		NO USE
LED04	数据库列不存在		NO USE
LED05	数据源连接失败	数据源连接失败	确认数据库链接信息正确且可访问
LED06	数据库备份执行失败		NO USE

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LED07	数据库清理执行失败		NO USE
LED08	数据库统计执行失败		NO USE
LED9	数据库正在备份, 不允许执行 DML 操作		NO USE
LED10	数据库正在恢复, 不允许执行任何操作	数据库正在恢复, 不允许执行任何操作	确认数据库恢复过程中无其他影响数据库的操作
LED11	快照已合并被清理, 请查看最新快照再操作	快照已合并被清理	确认最新快照, 根据需要进行后续操作, 或者修改快照合并规则
LED12	不存在 HA 或者分布式		NO USE
LED13	实例查询失败!	实例查询失败!	确认 EM 系统运行正常, 可稍后重试
LED14	实例修改失败!	实例修改失败!	根据修改失败提示进行相应操作
LED15	实例删除失败!	实例删除失败!	根据删除失败提示进行相应操作
LED16	实例添加失败!	实例添加失败!	根据添加失败提示进行相应操作
LED17	实例角色与实际部署不相符! 请选择正确的部署模式与实例角色!	实例角色与实际部署不相符	确认实例角色与实际部署要相同
LED18	实例角色检验失败!	实例添加, em 页面选择的实例角色与实例真实角色不同	执行命令获取实例真实角色
LED19	获取实例检验信息失败!	在实例所在主机执行 check_instance_param.sh 报错	1、em 日志查看执行 check_instance_param.sh 完整命令和失败原因 1.1 排查实例所在主机的主机用户/主机密码是否修改。 1.2 排查实例所在主机 check_instance_param.sh 脚本是否存在。
LED20	获取实例信息失败!	按照 ip 和端口从 em 数据库表 lem_instance 获取记录失败	在 em 数据库表 lem_instance 没有 ip 和端口对应的记录。
LED21	存在多个主节点!	高可用集群主节点已经被监控了	当前高可用集群主节点已经被监控了
LED22	数据库用户密码解密失败!	数据库密码 aes 解密失败	java 使用 em 类方法 com.hundsun.lightdb.utils.AesDecryptUtils#encryptS 来加密
LED23	主机代理已存在!	根据代理条件 (host_id,agent_port) 查询 lem_agent 表已存在	不必再添加主机代理
LED24	jdbc 直连被监控实例获取实例 DATA 目录失败	根据 ip, 端口, 数据库用户, 数据库密码, 利用 jdbc 执行 sql “select setting from pg_settings where name=’ data_directory’” 失败	1、排查 ip, 端口, 数据库用户, 数据库密码是否准确 2、使用数据库用户执行 sql “select setting from pg_settings where name=’ data_directory’” 是否报错
LED25	数据库实例安装目录没有权限	主机用户在数据库家目录 (LTHOME) 没有权限	使用在数据库家目录 (LTHOME) 有权限的主机用户

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LED26	数据库家目录无效	数据库家目录 (LTHOME) 是无效的	选择有效的数据库家目录 (LTHOME)
LED26	数据库实例 DATA 目录没有权限	主机用户在数据库实例 DATA 目录 (LTDATA) 没有权限	使用在数据库实例 DATA 目录 (LTDATA) 有权限的主机用户
LED27	代理表记录已存在!	根据代理条件 (host_id,agent_port) 查询 lem_agent 表已存在	不必再重复添加这个代理
LED28	实例信息无变化	代理状态正常, 数据库用户密码也没有变。	实例信息没有变化
LED29	主机实例下存在备份服务器定时任务, 请先删除定时任务!	根据实例 id(即 instance_id 对应 lem_instance 表主键) 查询 lem_backup_task(备份定时任务表) 记录不空。	在”集中备份»备份定时任务“页面删除实例的备份定时任务
LED30	主机实例下存在安装数据库目录, 请先删除安装数据库目录	实例是通过 em 在线安装的, 实例在 em 数据库表 lem_install_database(数据库安装表) 有记录	在主机管理页面, 选择主机记录, 执行卸载软件操作。
LED31	实例 id 不能为空	入参 instanceId(实例 id) 不能为空	
LED32	实例运行状态异常, 无法进行角色切换!	1、高可用集群主节点状态不正常 (不是 in production) 2、高可用集群备节点状态不正常 (不是 in archive recovery)	1、查看高可用集群中节点状态 (\$LTHOME/bin/lt_controldata -D \$LTDATA grep 'Database cluster state' cut -d ':' -f 2) 2、恢复高可用集群中状态异常的节点
LED33	获取实例运行状态脚本执行异常!	在实例所在主机执行获取实例运行状态的命令 (\$LTHOME/bin/lt_controldata -D \$LTDATA grep 'Database cluster state' cut -d ':' -f 2) 出现异常	1、检查主机 ip, ssh 端口, 主机用户, 主机密码是否准确。2、LTHOME(数据库家目录) 是否在主机存在且有效且当前主机用户有权限。3、LT-DATA(数据库实例 DATA 目录) 是否在主机存在且有效
LED34	角色切换失败!	在高可用集群执行实例角色切换命令 (sh \$LTHOME/bin/switchover.sh \$LTHOME) 失败	1、通过 em 日志或者角色切换的日志与进度, 查看实例角色切换报错的具体命令。2、根据步骤 1 报错的内容执行修复动作。
LED35	实例启动失败!	执行实例一键启停脚本, 启动失败	1、通过 em 日志或者一键启停的日志与进度, 查看执行一键启动失败的完整命令 (\$LTHOME/bin/lightdb_service.py -c start) 和失败原因 2、根据步骤 1 的报错内容, 修复后, 再执行一键启动。

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LED36	实例停止失败!	执行实例一键启停脚本, 停止失败	1、通过 em 日志或者一键启停的日志与进度, 查看执行一键停止失败的完整命令 (\$LTHOME/bin/lightdb_service.py -c stop) 和失败原因 2、根据步骤 1 的报错内容, 修复后, 再执行一键停止。
LED37	实例重启失败!	执行实例一键启停脚本, 重启失败	1、通过 em 日志或者一键启停的日志与进度, 查看执行一键重启失败的完整命令 (\$LTHOME/bin/lightdb_service.py -c restart) 和失败原因 2、根据步骤 1 的报错内容, 修复后, 再执行一键重启。
LED38	请至少选择一个节点!	在实例管理页面, 执行一键启停, 没有选择集群中的节点	执行实例启停, 选择集群中的节点
LED39	实例信息不存在!	按照实例 id(instanceId) 查询 em 数据库表 lem_instance 不存在。	实例信息已经不存在了
LED40	校验是否分布式高可用模式脚本执行异常!	执行命令 (if cat \$LTHOME/../../uninstall/uninstall grep 'clusterInsideDtoList') 校验实例的部署模式是否是分布式高可用失败。	1、通过 em 日志或者日志与进度查看报错的命令。2、查看实例的 uninstallFile.json 文件校验当前实例的部署模式是否是分布式高可用。3、检查实例所在主机 ip, 主机 ssh 端口, 主机用户和主机密码是否正确。
LED41	实例一键启停脚本执行异常!	执行实例一键启停脚本, 启动/停止/重启失败	1、通过 em 日志或者一键启停的日志与进度, 查看执行一键启停的完整命令 (\$LTHOME/bin/lightdb_service.py -c start) 和失败原因 2、根据步骤 1 的报错内容, 修复后, 再在实例管理页面执行一键启停。
LED42	实例端口不能为空!	入参实例端口 (instancePort) 不能为空	入参实例端口 (instancePort) 不能为空
LED43	角色一键切换脚本执行异常!	在高可用集群备节点执行实例角色切换命令 (sh \$LTHOME/bin/switchover.sh \$LTHOME) 失败	1、检查实例 (备节点) 所在主机的主机 ip, 主机 ssh 端口, 主机用户和主机密码是否正确。2、通过 em 日志或者角色切换的日志与进度, 查看实例角色切换报错的具体命令。3、根据步骤 1 报错的内容执行修复动作。
LED44	获取 HA 当前节点角色脚本执行异常!		NO USE
LED45	主节点角色切换未成功!		NO USE
LED46	备节点角色切换未成功!		NO USE

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LED47	实例节点不在监控范围!	根据主机 ip 和实例端口执行关联查询 (lem_instance li JOIN lem_host lh ON li.host_id = lh.host_id JOIN lem_cluster lc ON li.cluster_id = lc.cluster_id WHERE lh.host_ip = #{hostIp} and li.instance_port = #{instancePort}), 查询结果为空	根据主机 ip 和实例端口查看实例信息, 实例的主机信息和实例的集群信息是否存在。
LED48	实例安装目录不能为空	入参实例的安装目录不能为空	入参实例的安装目录不能为空
LED49	实例版本与集群中已有实例的数据库版本不一致	已被 em 监控的集群的实例数据库版本与待添加的集群实例数据库版本不一致。	查看待添加集群实例的数据库版本, 执行 Sql“SELECT version() as version”
LED50	代理不存在!	按照 hostIp 和实例 Port 条件执行关联查询 (FROM lem_agent la inner join lem_instance li on la.agent_id = li.agent_id where li.host_ip = #{hostIp} and li.instance_port = #{instancePort}), 实例的 agent 不存在	1、检查 hostIp 和实例端口是否在 em 数据库实例表 lem_instance 有记录 2、再检查步骤 1 实例的 agent_id 字段内容在 lem_agent 表是否有记录。
LED51	原生 PG 不支持实时 SQL!	原生 PG 未安装插件 pg_stat_activity 和插件 pg_stat_statements	为原生 pg 安装 lightdb 插件” lt_profile”, “lt_stat_activity”, “lt_stat_statements”, “lt_cron”
LED52	原生 PG 不支持 TOP,SQL!	原生 PG 未安装插件 pg_profile 和插件 pg_stat_statements	为原生 pg 安装 lightdb 插件” lt_profile”, “lt_stat_activity”, “lt_stat_statements”, “lt_cron”
LED53	原生 PG 不支持等待事件查询	原生 PG 未安装插件 pg_profile	为原生 pg 安装 lightdb 插件” lt_profile”, “lt_stat_activity”, “lt_stat_statements”, “lt_cron”
LED54	原生 PG 不支持 PWR 查询	原生 PG 未安装插件 pg_profile 和插件 pg_cron	为原生 pg 安装 lightdb 插件” lt_profile”, “lt_stat_activity”, “lt_stat_statements”, “lt_cron”
LED55	原生 PG 不支持 PWR 报告生成	原生 PG 未安装插件 pg_profile 和插件 pg_cron	为原生 pg 安装 lightdb 插件” lt_profile”, “lt_stat_activity”, “lt_stat_statements”, “lt_cron”
LED56	原生 PG 不支持 PWR 手动生成快照	原生 PG 未安装插件 pg_profile 和插件 pg_cron	为原生 pg 安装 lightdb 插件” lt_profile”, “lt_stat_activity”, “lt_stat_statements”, “lt_cron”
LED57	原生 PG 不支持 PWR 快照设置	原生 PG 未安装插件 pg_profile 和插件 pg_cron	为原生 pg 安装 lightdb 插件” lt_profile”, “lt_stat_activity”, “lt_stat_statements”, “lt_cron”
LED58	原生 PG 不支持实例巡检	原生 PG 不支持实例巡检	原生 PG 不支持实例巡检
LED59	添加实例, 实例所属数据库类型不支持	添加实例, 实例所属的数据库类型不在范围 [1:Lightdb, 2:PostgreSQL]	当前支持的数据库类型范围 [1:Lightdb, 2:PostgreSQL]

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LED60	添加实例，实例角色不支持	添加实例，实例角色不在范围 [Single Only, HA Primary, HA Standby, HA Witness, Distribute Coordinator Primary, Distribute Coordinator Standby, Distribute Coordinator Witness, Distribute Worker Primary, Distribute Worker Standby, Distribute Worker Witness]	添加实例，实例角色不在范围 [Single Only, HA Primary, HA Standby, HA Witness, Distribute Coordinator Primary, Distribute Coordinator Standby, Distribute Coordinator Witness, Distribute Worker Primary, Distribute Worker Standby, Distribute Worker Witness]
LED61	实例的数据库类型与集群的数据库类型不符	实例的数据库类型与集群的数据库类型不符	实例的数据库类型与集群的数据库类型不符
LED62	jdbc 连接被监控实例获取数据库版本失败	通过 jdbc 连接被监控实例执行 SQL(“SELECT version() as version;”) 获取数据库版本失败	1、排查 ip, 端口, 数据库用户, 数据库密码是否准确 2、使用数据库用户执行 sql “SELECT version() as version;” 是否报错
LED63	被监控原生 PG 实例的数据库版本小于 13	被监控原生 PG 实例的数据库版本小于 13	EM 不支持原生 PG 版本小于 13
LED64	密码安全性较弱! 请修改密码, 密码必须包含数字、小写字母、大写字母、特殊字符 (@#%&+=), 并且长度至少为 8 个字符!	密码安全性较弱! 请修改密码, 密码必须包含数字、小写字母、大写字母、特殊字符 (@#%&+=), 并且长度至少为 8 个字符!	密码必须包含数字、小写字母、大写字母、特殊字符 (@#%&+=), 并且长度至少为 8 个字符!
LEA01	告警不存在		NO USE
LEA02	待办告警不存在		NO USE
LEA03	告警通知方式设置异常		NO USE
LEA04	告警阈值格式校验失败		NO USE
LEA05	告警内部业务异常, //告警内部业务处理时出现异常		NO USE
LEA06	待办告警处理失败	未查询到待处理的告警 (根据 id 查询表 sys_task)	未查询到待处理的告警 (根据 id 查询表 sys_task)
LEA07	待办告警未选择或确认信息为空	处理待办任务, 未选择待办告警或者确认信息为空	处理待办任务, 未选择待办告警或者确认信息为空
LEM01	任务周期格式校验异常, //		NO USE
LEM02	任务创建失败, //		NO USE
LEM03	任务修改失败, //		NO USE
LEM04	任务查询失败, //		NO USE
LEM05	任务删除失败, //		NO USE
LEM06	任务执行失败, //		NO USE
LEM07	任务不存在, //		NO USE
LEM08	任务日志不存在, //		NO USE
LEL01	无锁详情		NO USE
LEL02	分布式锁释放失败!	某些业务操作比如删除主机 会释放分布式锁,	redis 进程不存在了, 在 em 所在服务器检查 redis 进程是否存在
LEL03	该集群正在角色切换或实例启停, 请稍后再试!	1、集群正在角色切换 2、集群正在实例启停	稍后再试, 等待集群角色切换完成或者等待实例启停完成

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEB01	上线失败!	备份定时任务添加到调度失败	1、查看 em 日志查看备份定时任务添加到调度失败的异常信息。2、在集中备份 » 备份定时任务页面, 重新执行备份定时任务上线。
LEB02	下线失败!	备份定时任务移除失败	1、备份定时任务已经在执行了无法被移除。2、查看 em 日志中记录的备份定时任务移除失败的日志信息。
LEB03	获取存储阈值信息失败!	执行命令 (df -B 1g \$backup_directory awk '{print \$2,\$4,\$5}' sed -n '2p') 判断备份服务器上备份目录的可用空间大小 (GB) 失败	查看备份服务器 ip, 备份服务器用户, ssh 端口, 备份服务器用户密码是否准确。
LEB04	备份目录未初始化!	备份定时任务执行, 备份服务器上备份目录没有初始化。	重新在集中备份 » 备份定时任务页面添加备份定时任务
LEB05	执行备份启动脚本失败!	执行单机, 高可用主节点, 或者分布式备份失败。	不论分布式备份失败, 还是单机, 高可用主节点备份失败, 1、在备份定时任务启动过程中, 查看日志与进度输出的失败信息并修复。2、在备份服务器的备份目录下的备份日志 (<code>\${backup_dir}/backup_logs/\${instance_name}/back</code>) 查看单机, 高可用主节点备份失败日志信息。3、在备份服务器查看分布式备份日志文件 (<code>\${backup_dir}/log/lt_distributed_probackup-%Y-%m-%d.log</code>) 中分布式备份失败信息并修复。
LEB06	执行备份清理脚本失败!	1、执行命令 (lt_probackup delete) 清理过期备份数据失败 2、执行命令 (lt_probackup delete) 按照备份 id 删除数据失败	1、查看删除过期备份数据日志文件 (<code>\${backup_dir}/backup_logs/\${instance_name}/back</code>) 内容并修复 2、查看按照备份 id 删除备份数据日志文件 (<code>\${backup_dir}/backup_logs/\${instance_name}/back</code>) 内容并修复。
LEB07	执行备份列表脚本失败!	查看单机, 高可用或者是分布式备份列表失败。	1、查看备份服务器 ip, 备份服务器 ssh 端口, 备份服务器用户, 备份服务器用户密码是否变化。2、查看备份服务器上备份目录是否还存在。3、查看备份服务器上备份目录下备份实例目录是否存在。

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEB08	执行备份列表详情脚本失败!	查看单机, 高可用或者是分布式备份列表中某次备份 id 的详情失败。	1、查看备份服务器 ip, 备份服务器 ssh 端口, 备份服务器用户, 备份服务器用户密码是否变化。2、查看备份服务器上备份目录是否还存在。3、查看备份服务器上备份目录/备份实例目录是否存在。4、查看备份服务器上备份目录/备份实例目录/某次备份 id 目录是否存在。
LEB09	备份服务器为空!	根据 ip 无法在 em 数据库备份服务器表 lem_backup_server 找到记录。	备份服务器需要重新添加。
LEB10	备份服务器列表数据解析失败!	备份实例的备份列表查询, 返回的行列数据解析失败。	在备份服务器查询备份实例的备份列表并解析。
LEB11	备份服务器列表详情数据解析失败!	备份实例的备份列表中备份 id 的详情查询, 返回的行列数据解析失败。	备份实例的备份列表中备份 id 的详情查询并解析。
LEB12	工具 lt_probackup 的目录在备份服务器不存在	lt_probackup 的目录在备份服务器不存在	在备份服务器定位 lt_probackup 所在目录绝对路径
LEB13	备份目录已经初始化过了	备份服务器上备份目录已经完成了初始化	备份服务器上备份目录已经完成了初始化, 不必再次初始化, 或者更换别的备份目录。
LEB14	当前用户没有权限操作该备份目录	主机用户在备份服务器上没有备份目录操作权限	选择备份服务器上主机用户有权限操作的目录作为备份目录。
LEB15	当前用户不能创建该备份目录	当前用户不能在备份服务器上创建该备份目录	更换一个有权限的备份目录。
LEB16	为备份目录添加实例失败	在备份服务器的备份目录添加实例失败 (lt_probackup add-instance)	1、若是在备份服务器的备份目录为远程主机上实例添加备份, 备份服务器与远程实例所在服务器没有 ssh 相互免密。2、备份服务器上备份目录不存在。3、备份实例的 DATA 目录在主机上不存在。
LEB17	为 lt_probackup 添加配置失败	为 lt_probackup 添加配置失败	1、若是在备份服务器的备份目录为远程主机上实例添加备份, 备份服务器与远程实例所在服务器没有 ssh 相互免密。2、备份服务器上备份目录不存在。3、备份实例的 DATA 目录在主机上不存在。4、备份实例进程在主机上不存在。5、备份实例不存在数据库用户 lt_probakup
LEB18	备份定时任务查询失败	根据备份定时任务 id 无法在 em 数据库备份定时任务表 lem_backup_task 查询到记录	备份定时任务已被删除, 需要重新添加备份定时任务

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEB19	备份定时任务在线不允许删除	备份定时任务在线不允许删除	备份定时任务在线不允许删除，需要先下线再删除。
LEB20	备份定时任务删除失败	根据备份定时任务 id 删除 em 数据库备份定时任务表 lem_backup_task 记录失败	em 数据库已经崩了
LEB21	备份目录查询失败	根据备份目录 id 无法在 em 数据库备份目录表 lem_backup_directory 查询到记录	备份目录记录在 em 数据库备份目录表 lem_backup_directory 已经被删除了
LEB22	备份日志删除失败！		NO USE
LEB23	备份定时任务 id 不能为空	入参备份定时任务 id 不能为空	入参备份定时任务 id 不能为空
LEB24	备份定时任务上下线不能为空	入参定时任务上下线 (cronOnline) 不能为空	入参定时任务上下线 (cronOnline) 不能为空
LEB25	备份定时任务上下线值不合法	入参定时任务上下线 (cronOnline) 内容不合法	入参定时任务上下线 (cronOnline) 内容范围 [0: 下线, 1: 上线]
LEB26	备份定时任务备份失败！		NO USE
LEB26	备份服务器 id 不能为空	备份服务器 id 不能为空	确保备份服务器 id 不能为空
LEB27	不能删除，需清空备份服务器的所有备份定时任务	不能删除，需清空备份服务器的所有备份定时任务	确保清空备份服务器的所有备份定时任务才能删除备份服务器
LEB28	备份定时任务添加失败	备份定时任务添加失败	插入表记录到备份目录表 lem_backup_task 失败
LEB29	获取备份列表为空！	获取备份列表为空！	确保执行备份的时候备份服务器的备份目录的备份实例下的脚本 backup_list.sh 执行不为空
LEB30	获取备份列表详情为空！	获取备份列表详情为空！	确保执行备份的时候备份服务器上备份脚本 backup_listDetail.sh 执行不为空
LEB31	该备份服务器上面有跑着数据库实例，不能在该服务器上添加备份定时任务！	该备份服务器上面有跑着数据库实例，不能在该服务器上添加备份定时任务！	确保备份服务器上面有跑着数据库实例，不在该服务器上添加备份定时任务
LEB32	备份服务器已存在	备份服务器已存在	备份服务器已存在，则不重复添加
LEB33	主机实例下存在备份服务器定时任务，请先删除定时任务！	主机实例下存在备份服务器定时任务，请先删除定时任务！	删除主机的时候确保该主机下没有备份服务器定时任务
LEB34	备份服务器 ip 不能为空	备份服务器 ip 不能为空	备份服务器 ip 不能为空
LEB35	备份服务器 ip 不合法	备份服务器 ip 不合法	确保备份服务器 ip 合法
LEB36	备份服务器用户不能为空	备份服务器用户不能为空	备份服务器用户不能为空
LEB37	lt_probackup 路径不能为空	lt_probackup 路径不能为空	lt_probackup 路径不能为空
LEB38	ssh 端口不能为空	ssh 端口不能为空	ssh 端口不能为空
LEB39	剩余空间大小不能为空	剩余空间大小不能为空	剩余空间大小不能为空
LEB40	剩余空间占比不能为空	剩余空间占比不能为空	剩余空间占比不能为空

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEB41	获取 lt_probackup 工具版本为空	获取 lt_probackup 工具版本为空	查看备份服务器上备份脚本目录 (backup) 所在目录 (backup_scripts) 下 lt_probackup_version.sh 的执行结果是否为空
LEB42	备份服务器用户密码不能为空	备份服务器用户密码不能为空	备份服务器用户密码不能为空
LEB43	lt_probackup 路径下没有工具 lt_probackup	lt_probackup 路径下没有工具 lt_probackup	lt_probackup 路径下没有工具 lt_probackup
LEB44	备份并行数不能为空	备份并行数不能为空	备份并行数不能为空
LEB45	备份并行数不合法, 范围 [1,64]	备份并行数不合法, 范围 [1,64]	备份并行数不合法, 范围 [1,64]
LEB46	备份周期天数不合法, 范围 [7,90]	备份周期天数不合法, 范围 [7,90]	备份周期天数不合法, 范围 [7,90]
LEB47	备份用户不能为空	备份用户不能为空	备份用户不能为空
LEB48	备份用户只支持 lt_probackup	备份用户只支持 lt_probackup	备份用户只支持 lt_probackup
LEB49	备份每天执行时间不能为空	备份每天执行时间不能为空	备份每天执行时间不能为空
LEB50	备份每天执行时间格式不合法, 格式为 HH:mm	备份每天执行时间格式不合法, 格式为 HH:mm	备份每天执行时间格式不合法, 格式为 HH:mm
LEB51	备份存储目录不能为空	备份存储目录不能为空	备份存储目录不能为空
LEB52	备份数据库实例 id 不能为空	备份数据库实例 id 不能为空	备份数据库实例 id 不能为空
LEB53	增量备份类型不能为空	增量备份类型不能为空	增量备份类型不能为空
LEB54	增量备份类型目前只支持取值为 DELTA,PAGE	增量备份类型目前只支持取值为 DELTA,PAGE	增量备份类型目前只支持取值为 DELTA,PAGE
LEB55	压缩算法不能为空	压缩算法不能为空	压缩算法不能为空
LEB56	不选择压缩算法 (none), 压缩级别只能为 0	不选择压缩算法 (none), 压缩级别只能为 0	确保选择了压缩算法
LEB57	该数据库实例已有备份定时任务无需再添加	该数据库实例已有备份定时任务无需再添加	该数据库实例已有备份定时任务无需再添加
LEB58	本地数据库实例备份, 配置文件 postgresql 配置项 archive_command 内容需要去除 remote,option	本地数据库实例备份, 配置文件 postgresql 配置项 archive_command 内容需要去除 remote,option	确保本地数据库实例备份, 配置文件 postgresql 配置项 archive_command 内容需要去除 remote,option
LEB33	该实例下不存在 OK 的全量备份, 请先执行全量备份, 后续可执行增量备份!	该实例下不存在 OK 的全量备份, 请先执行全量备份, 后续可执行增量备份!	确保实例下存在 OK 的全量备份, 然后再执行增量备份
LEB59	远程实例备份, 配置文件 postgresql.conf 中配置项 archive_command 的内容需要 remote,option	远程实例备份, 配置文件 postgresql.conf 中配置项 archive_command 的内容需要 remote,option	确保远程实例备份, 配置文件 postgresql.conf 中配置项 archive_command 的内容需要 remote,option
LEB60	需要在 \$LTDATA/lt_hba.conf 配置备份用户 lt_probackup 的访问权限和免密	需要在 \$LTDATA/lt_hba.conf 配置备份用户 lt_probackup 的访问权限和免密	在 \$LTDATA/lt_hba.conf 配置备份用户 lt_probackup 的访问权限和免密
LEB61	备份服务器上工具 lt_probackup 版本与数据库实例工具 lt_probackup 版本不同	备份服务器上工具 lt_probackup 版本与数据库实例工具 lt_probackup 版本不同	确保备份服务器上工具 lt_probackup 版本与数据库实例工具 lt_probackup 版本相同
LEB62	执行备份目录初始化失败	执行备份目录初始化失败	手动执行备份实例下的 make_backup_directory_initial.sh, 查看报错日志

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEB163	待初始化的目录不空	待初始化的目录不空	确保待初始化的目录为空
LEB34	手动执行备份失败!	手动执行备份失败!	请检查日志和相关实例!
LEB63	备份存储目录不是 linux 绝对路径, 路径支持.-_ 数字和英文字母大小写	备份存储目录不是 linux 绝对路径, 路径支持.-_ 数字和英文字母大小写	确保备份存储目录是 linux 绝对路径, 路径格式是.-_ 数字和英文字母大小写
LEB99	备份未知异常	备份未知异常	请检查日志和相关实例!
LER101	备份删除记录不存在	备份删除记录不存在	确保备份删除记录存在
LER102	备份任务不存在	备份任务不存在	请检查日志和相关实例!
LER103	备份服务器密码解密失败	备份服务器密码解密失败	确保备份服务器密码未被修改
LEB104	备份删除策略为 [3-指定备份状态删除] 但备份状态为空	备份删除策略为 [3-指定备份状态删除] 但备份状态为空	确保备份删除策略为 [3-指定备份状态删除] 但备份状态不为空
LEB105	备份删除策略为 [4-指定备份记录删除] 但备份 ID 为空	备份删除策略为 [4-指定备份记录删除] 但备份 ID 为空	确保备份删除策略为 [4-指定备份记录删除] 但备份 ID 不为空
LEB106	备份删除入参并行线程数不合法 (允许范围为 [1,64])	备份删除入参并行线程数不合法 (允许范围为 [1,64])	确保备份删除入参并行线程数合法 (允许范围为 [1,64])
LEB107	备份删除入参状态暂不支持 (允许状态为,CORRUPT、ERROR、ORPHAN)	备份删除入参状态暂不支持 (允许状态为,CORRUPT、ERROR、ORPHAN)	确保入参状态为,CORRUPT、ERROR、ORPHAN
LEB108	搜索项【备份服务器/备份目录/备份实例】不能为空	搜索项【备份服务器/备份目录/备份实例】不能为空	搜索项【备份服务器/备份目录/备份实例】不能为空
LEB109	在备份服务器执行检查 lt_probackup 是否存在的命令出错, 添加备份服务器失败	在备份服务器执行检查 lt_probackup 是否存在的命令出错, 添加备份服务器失败	请检查日志和相关实例!
LEB110	lt_probackup 在备份服务器不存在, 添加备份服务器失败	lt_probackup 在备份服务器不存在, 添加备份服务器失败	检查 lt_probackup 在备份服务器是否存在
LEB111	备份服务器上 lt_probackup 版本与用于备份的数据库实例的 lt_probackup 版本不同	备份服务器上 lt_probackup 版本与用于备份的数据库实例的 lt_probackup 版本不同	请检查日志和相关实例!
LEB112	在主机备份实例 DATA 目录执行脚本 modify_para_continuous_wal_archivin 执行失败	在主机备份实例 DATA 目录执行脚本 modify_para_continuous_wal_archivin 执行失败	
LEB113	备份服务器通过 ssh 协议连接不上	备份服务器通过 ssh 协议连接不上	检查网络以及服务器的用户名密码是否正确
LEB114	恢复实例启动失败	恢复实例启动失败	请检查日志和相关实例!
LEB115	手工重启恢复实例	手工重启恢复实例	手工重启恢复实例
LEB116	初始化备份服务器备份目录失败	初始化备份服务器备份目录失败	手动执行备份服务器下的 belt_distributed_probackup.py 查看报错原因
LEB117	在备份服务器备份目录添加分布式集群实例信息失败	在备份服务器备份目录添加分布式集群实例信息失败	请检查日志和相关实例!
LEB118	在分布式集群节点设置持续 WAL 归档失败	在分布式集群节点设置持续 WAL 归档失败	请检查数据库日志查看归档失败原因
LEB119	在分布式集群执行备份失败	在分布式集群执行备份失败	请检查日志和相关实例!
LEB120	恢复后启动, 恢复后的实例必须为 CN 主节点	恢复后启动, 恢复后的实例必须为 CN 主节点	确保恢复后启动, 恢复后的实例为 CN 主节点

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEB121	分布式集群恢复, Lightdb 数据库版本需要 >=22.4	分布式集群恢复, Lightdb 数据库版本需要 >=22.4	确保分布式集群恢复, Lightdb 数据库版本需要 >=22.4
LEB122	分布式集群 cn 节点下面没有 dn 节点	分布式集群 cn 节点下面没有 dn 节点	请检查日志和相关实例!
LEB123	备份实例与恢复实例需要均为分布式集群 cn 主节点	备份实例与恢复实例需要均为分布式集群 cn 主节点	确保备份实例与恢复实例均为分布式集群 cn 主节点
LEB124	分布式集群备份配置 lt_probackup 失败	分布式集群备份配置 lt_probackup 失败	请检查日志和相关实例!
LEB125	暂不支持分布式备份	暂不支持分布式备份	NO USE
LEB126	原生 PG 不支持集中备份	原生 PG 不支持集中备份	原生 PG 不支持集中备份
LEB127	恢复实例的 DATA 目录不空	恢复实例的 DATA 目录不空	确保恢复实例的 DATA 目录初始为空
LEB128	主机用户没有权限创建恢复实例的 DATA 目录	主机用户没有权限创建恢复实例的 DATA 目录	确保主机用户有权限创建恢复实例的 DATA 目录
LEB129	分布式集群节点数量不一致	分布式集群节点数量不一致	分布式集群节点数量不一致
LEB130	恢复后 CN 节点更新恢复后端口失败	恢复后 CN 节点更新恢复后端口失败	NO USE
LEB131	备份实例需要为分布式集群 cn 主节点	备份实例需要为分布式集群 cn 主节点	确保备份实例为分布式集群 cn 主节点
LEB132	清除 CN 节点的元数据信息	清除 CN 节点的元数据信息	查看实例日志执行 truncate pg_dist_node; 失败的原因
LEB133	设置恢复后 CN 节点的元数据信息	设置恢复后 CN 节点的元数据信息	设置恢复后 CN 节点的元数据信息
LEB134	设置恢复后 DN 节点的元数据信息	设置恢复后 DN 节点的元数据信息	设置恢复后 DN 节点的元数据信息
LEI01	LightDB 安装包命名不规范, 请使用初始安装包名!	LightDB 安装包命名不规范, 请使用初始安装包名!	确保 LightDB 安装包命名不规范, 请使用初始安装包名!
LEI02	LightDB 安装包上传失败, 存储磁盘可用容量小于 3G	LightDB 安装包上传失败, 存储磁盘可用容量小于 3G	确保存储磁盘可用容量大于 3G
LEI03	LightDB 安装包上传失败	LightDB 安装包上传失败	请检查日志和相关实例!
LEI04	LightDB 安装包不存在, 请重新上传	LightDB 安装包不存在, 请重新上传	确保 LightDB 安装包存在
LEI05	路径存在文件或目录, 请更换路径!	路径存在文件或目录, 请更换路径!	路径存在文件或目录, 请更换路径!
LEI06	路径无权限!	路径无权限!	确保路径有权限
LEI07	无法获取远程主机 CPU 和操作系统信息, LightDB 无法适配远程主机!	无法获取远程主机 CPU 和操作系统信息, LightDB 无法适配远程主机!	请检查日志和相关实例!
LEI08	LightDB 安装包无法适配远程主机 CPU 和操作系统, 请重新上传或选择正确的安装包!	LightDB 安装包无法适配远程主机 CPU 和操作系统, 请重新上传或选择正确的安装包!	确保选择正确的安装包!
LEI09	无法获取该路径可用存储磁盘空间大小!	无法获取该路径可用存储磁盘空间大小!	无法获取该路径可用存储磁盘空间大小!
LEI10	LightDB 安装检测系统参数错误, 请阅读最新的安装手册并设置正确的系统参数!	LightDB 安装检测系统参数错误, 请阅读最新的安装手册并设置正确的系统参数!	LightDB 安装检测系统参数错误, 请阅读最新的安装手册并设置正确的系统参数!
LEI11	LightDB 安装检测未安装的依赖包, 请阅读最新的安装手册并安装正确的依赖包!	LightDB 安装检测未安装的依赖包, 请阅读最新的安装手册并安装正确的依赖包!	LightDB 安装检测未安装的依赖包, 请阅读最新的安装手册并安装正确的依赖包!

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEI12	路径可用存储磁盘空间较小, 无法继续操作, 请检查磁盘空间!	路径可用存储磁盘空间较小, 无法继续操作, 请检查磁盘空间!	路径可用存储磁盘空间较小, 无法继续操作, 请检查磁盘空间!
LEI13	无法获取系统依赖包, 请检查远程主机状态!	无法获取系统依赖包, 请检查远程主机状态!	无法获取系统依赖包, 请检查远程主机状态!
LEI14	无法获取系统参数, 请检查远程主机状态!	无法获取系统参数, 请检查远程主机状态!	无法获取系统参数, 请检查远程主机状态!
LEI15	; 系统依赖包未安装:	; 系统依赖包未安装:	安装系统依赖包
LEI16	; 系统参数设置有误:	; 系统参数设置有误:	修改系统参数设置
LEI17	正在安装检测中, 请稍后重试!	对同一个 ip 和目录同一时刻无法多次安装数据库	等该主机和目录下的数据库安装完毕之后再安装
LEI18	请求参数缺少, 请检查接口参数!	数据库安装的信息封装的实体类信息缺失	按照要求添加数据库安装等信息
LEI19	LightDB 安装目录缺少, 请填写 LightDB 安装目录缺少!	数据库安装的目录等封装的实体类信息缺失	按照要求添加数据库安装的安装目录等信息
LEI20	缺少 ip 信息, 请选择主机!	数据库安装的主机的 ip 信息丢失	数据库安装的主机的 ip 信息丢失, 要选择正确的主机
LEI21	主机信息缺失, 请先添加主机!	数据库安装的主机的其他信息丢失	数据库安装的主机的其他信息丢失, 选择正确的主机
LEI25	无法获取集群机器操作系统和 CPU 信息, 请检查 SSH 连接!	无法获取集群机器操作系统和 CPU 信息, 请检查 SSH 连接!	无法通过 ssh 连接远程主机, 获取远程主机的信息, 请检查网络通信、防火墙等是否正常
LEI26	LightDB 安装包无法适配集群机器, 请上传正确安装包!	LightDB 安装包无法适配集群机器	请上传正确 LightDB 安装包!
LEI30	LightDB 安装包解压失败, 请检查磁盘空间等!	安装包是个压缩包, 需要在机器上面解压, em 的目录空间不足	保证 em 的工作目录磁盘容量充足
LEI31	LightDB 安装包安装失败, 请检查磁盘空间等!	安装 lightDB 需要一定的磁盘空间, 而实际目录的磁盘容量不足	保证 LightDB 的安装目录磁盘容量充足
LEI35	LightDB 实例路径空间不足, 请检查磁盘空间等!	实例化 LightDB 需要一定的磁盘空间, 而实际目录的磁盘容量不足	保证实例化 LightDB 的目录磁盘容量充足
LEI36	LightDB 实例初始化失败, 请检查内存!	LightDB 实例初始化失败, 实例化机器的可用内存不足	保证实例化 LightDB 的机器可用内存充足
LEI50	无法校验远程机器字符集, 请检查远程机器数据库软件!	获取实例化 LightDB 的机器字符集失败	检查远程主机网络连接是否正常
LEI51	数据库只支持 UTF8, GBK, 请更换机器!	LightDB 支持的机器字符为 UTF8, GBK	这个主机不支持安装 LightDB, 换个主机
LEI56	无法获得机器端口信息, 请检查远程机器!	LightDB 安装的机器端口信息获取不到	检查远程主机网络连接是否正常
LEI57	无法获得可用端口, 请检查远程机器!	安装的主机机器无可用的端口	释放被占用的主机端口
LEI60	无法获取主机内存信息!	无法获取内存信息	保证远程主机连接信息正常
LEI65	主机数据库实例化失败, 请查看日志!	数据库实例化失败	查看实例的日志
LEI66	主机数据库实例化设置参数失败, 请查看日志!	数据库实例化失败	查看实例的相关日志
LEI70	主机数据库实例启动失败, 请查看具体日志!	数据库实例启动失败	查看实例的运行日志

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LEI71	主机数据库实例安装插件失败, 请查看具体日志!	数据库实例安装插件失败	查看实例的相关日志
LEI72	主机数据库 hba 设置错误, 请查看具体日志!	数据库安装的 hba 设置错误	查看实例安装的相关日志
LEI80	集群类型仅支持在线安装, 请重新选择!	集群类型仅支持在线安装, 请重新选择!	选择在线安装
LEI81	部署模式暂只支持单机部署和高可用部署, 敬请期待!	部署模式暂只支持单机部署和高可用部署	部署模式暂只支持单机部署和高可用部署
LEI82	单机部署模式暂只支持单个实例, 请重新选择填写!	单机部署模式暂只支持单个实例	单机部署模式暂只支持单个实例
LEI83	请选择主机 IP!	未选择主机 IP	选择主机 IP
LEI84	保存实例信息失败, 请查看日志!	保存实例信息失败	查看实例的日志
LEI90	正在实例化中, 请稍后重试!	正在实例化中	实例化完毕, 再重试!
LEI91	生成 uninstallFile 文件失败!	生成卸载文件失败	查看实例安装的相关日志
LEI92	安装数据库的版本不能小于 22.3	安装数据库的版本小于 22.3	选择大于 22.3 版本的 LightDB 安装包
LEU01	请选择卸载节点等信息!	未选择卸载节点的信息	选择卸载节点等信息
LEU02	实例存在 Agent, 请先删除实例 Agent!	卸载实例请先卸载 agent	请先删除实例 Agent!
LEU03	实例的 LTHOME 信息丢失, 请先维护对应的数据库软件信息!	实例的 LTHOME 信息丢失	请先维护对应的数据库软件信息!
LEU04	实例停止失败, 请检查日志和相关实例!	实例停止失败	请检查日志和相关实例!
LEU04	实例删除失败, 请检查日志和相关实例!	实例删除失败	请检查日志和相关实例!
LEU05	实例的 ltcluster 停止失败, 请检查日志和相关实例!	实例的 ltcluster 停止失败	请检查日志和相关实例!
LEU10	数据库存在其他实例, 请先将其他实例删除!	数据库存在其他实例	请先将其他实例删除!
LEU011	数据库删除失败, 请检查日志和相关实例!	数据库删除失败	请检查日志和相关实例!
LEU012	缺少 installDatabaseId 信息	缺少 installDatabaseId 信息	维护存储的 installDatabaseId 信息
LEU013	缺少 hostIP 信息	缺少 hostIP 信息	维护存储的 hostIP 信息
LEU014	缺少 ltBase 信息	缺少 ltBase 信息	维护存储的 ltBase 信息
LEU015	待卸载实例运行状态异常	待卸载实例运行状态异常	维护实例运行状态正常
LER01	无效的恢复方式 (恢复方式: 1-基于备份, 2-基于时间点)	恢复方式不对	选择以下恢复方式: 1-基于备份, 2-基于时间点
LER02	恢复方式为 [1-基于备份] 但传入的备份 ID 为空	传入的备份 ID 为空	传入正确的备份 ID 为空
LER03	恢复方式为 [2-基于时间点] 但传入的时间点为空	传入的时间点为空	传入正确的时间点
LER04	入参备份实例不存在	备份实例不存在	维护正确的备份实例信息
LER05	入参恢复实例不存在	恢复实例不存在	维护正确的恢复实例信息
LER06	备份服务器主机密码解密失败	备份服务器主机密码解密失败	密码错误, 维护正确的加密的密码信息
LER07	SSH 连接失败	SSH 连接失败	检查主机网络信息、防火墙信息

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
LER08	恢复实例主机密码解密失败	恢复实例主机密码解密失败	密码错误, 维护正确的加密的密码信息
LER09	备份服务器和恢复实例服务器 ntp 一致性校验失败	备份服务器和恢复实例服务器 ntp 一致性校验失败	查看集群 ntp 是否正常
LER10	备份或还原服务器 ntp 异常	备份或还原服务器 ntp 异常	查看集群 ntp 是否正常
LER11	备份服务器和恢复实例服务器 ntp 一致性校验失败	备份服务器和恢复实例服务器 ntp 一致性校验失败	查看集群 ntp 是否正常
LER15	恢复实例,probackup, 版本获取失败	恢复实例,probackup, 版本获取失败	检查 probackup 版本等
LER16	备份服务器和恢复实例,probackup, 版本不一致	备份服务器和恢复实例,probackup, 版本不一致	保持备份服务器和恢复实例,probackup, 版本一致
LER17	恢复添加入参并行线程数不合法 (允许范围为 [1,64])	恢复添加入参并行线程数不合法	并行线程数目为允许范围为 [1,64]
LER18	备份实例与恢复实例的数据库版本不一致	备份实例与恢复实例的数据库版本不一致	保证备份实例与恢复实例的数据库版本一致
LER14	指定的还原备份记录不存在	指定的还原备份记录不存在	保证还原备份记录存在
LPR01	新增日志与进度记录失败	新增日志与进度记录失败	检查相应的日志是否存在、存储空间是否够用
LPR02	更新日志与进度记录失败	更新日志与进度记录失败	检查相应的日志是否存在、存储空间是否够用
LEH1	redis 获取的启动 keepalived 和 ntp 命令为空	redis 获取的启动 keepalived 和 ntp 命令为空	缓存丢失了
LEH2	启动 keepalived 和 ntp 命令失败	启动 keepalived 和 ntp 命令失败	手动启动 keepalived 和 ntp
LEK1	获取告警配置信息失败	获取告警配置信息失败	检查网络等
LEM1	修改配置文件告警信息失败	修改配置文件告警信息失败	检查网络
LEM2	邮箱账户格式不正确	邮箱账户格式不正确	配置正确的邮箱账号
LEM3	邮箱服务器配置不正确	邮箱服务器配置不正确	配置邮箱服务器配置
LEM4	邮箱告警配置错误或网络不通	邮箱告警配置错误或网络不通	检查网络和防火墙
LEM5	钉钉告警配置错误或网络不通	钉钉告警配置错误或网络不通	检查网络和防火墙
LEM6	配置文件不存在	配置文件不存在	检查代理的配置文件
LEM7	agentType 类型只能是 instance 或者 host	agentType 类型只能是 instance 或者 host	仅仅选择 instance 或者 host
LEM8	钉钉参数配置不全	钉钉参数配置不全	保证钉钉参数正确
LEM9	邮箱参数配置不全	邮箱参数配置不全	保证邮箱参数配置正确
LEJ1	一键确认所有告警信息失败	一键确认所有告警信息失败	重试
LEJ10	备份服务器的版本应该大于等于 13.8-22.4	备份服务器的版本应该大于等于 13.8-22.4	检查备份服务器的版本信息
LEK10	data 输入为空	data 输入为空	保证 data 信息正确
EK11	dbName 为空	dbName 为空	保证 dbName 信息正确
EK12	tableName 为空	tableName 为空	保证 tableName 信息正确
EK13	sqlContent 为空	sqlContent 为空	保证 sqlContent 信息正确
EK14	manualProcessResult 为空	manualProcessResult 为空	保证 manualProcessResult 信息正确
EK15	审核工单内容重复, 一分钟内不能重复提交, 请稍后再试	审核工单内容重复, 一分钟内不能重复提交, 请稍后再试	一分钟后重试

续下页

表 3 - 接上页

错误码	中文描述	原因	解决方法
EK16	工单已被人工审核, 请勿重复审核	工单已被人工审核, 请勿重复审核	请勿重复审核
EK17	该 id 不存在工单	该 id 不存在工单	保证该 id 信息正确
EK18	clusterId 不能为空	clusterId 不能为空	保证 clusterId 信息为空
EK19	查询实例拥有的数据库失败, 请检查数据库实例和数据库实例代理是否正常	查询实例拥有的数据库失败	请检查数据库实例和数据库实例代理是否正常
EK20	SQL 文本字符长度超过 4000	SQL 文本字符长度超过 4000	保证 SQL 文本字符长度不超过 4000
EK21	数据库名称字符长度超过 100	数据库名称字符长度超过 100	保证数据库名称字符长度不超过 100
EK22	分隔符为单个字符, 不能为空, 且分隔符不支持.\$IO[{}^?*\类型	分隔符为单个字符, 不能为空, 且分隔符不支持.\$IO[{}^?*\类型	分隔符为单个字符, 不能为空, 且分隔符不支持.\$IO[{}^?*\类型
EK23	EXPLAIN 失败, 请检查 SQL	EXPLAIN 失败	请检查 SQL
LEK24	重载数据库配置文件失败	重载数据库配置文件失败	查看数据库配置文件, 查看数据库日志
LEK25	入参不合法, 包含不可修改的 guc 参数	入参不合法, 包含不可修改的 guc 参数	查看数据库文档, 检查参数是否合法
LEK26	guc 参数修改失败	guc 参数修改失败	查看数据库文档, 检查参数是否合法
LEK27	部分 guc 参数修改失败, 且恢复到原参数值失败	部分 guc 参数修改失败, 且恢复到原参数值失败	查看数据库文档, 检查参数是否合法
LEK28	guc 参数集合为空	guc 参数集合为空	保证正确的 guc 参数集合

8 版本发布历史

8.1 版本发布时间线

本节列出了所有已发布的 LightDB-EM 版本, 按发布时间倒序呈现。

LightDB-EM 版本	发布日期
23.4	2023-12-29
23.3	2023-09-30
23.2	2023-06-30
23.1	2023-03-31
22.4	2022-12-31
22.3	2022-09-30

8.2 发布记录

LightDB-EM 23.4 版本升级内容

1. 基于 em13.8-23.3 版本，添加兼容 22.1 版本 lightdb 的备份与恢复；
2. EM-记录所有操作日志；
3. EM-SQL 上线支持任意数据库；
4. 高可用集群和分布式常规模式下所有 Witness 节点不校验 keepalived 状态；
5. 支持在线创建租户（选择 oracle/mysql 模式）；
6. 在线创建数据库支持选择字符集编码，并可支持开发者模式；
7. EM 在线部署 lightdb 实例增加开发者选项和增量检查点；

LightDB-EM 23.3 版本升级内容

1. 主机代理支持同时监控数据库实例，最多监控 4 个数据库实例，便于资源的合理利用，减少管理的复杂性。
2. 数据库日志中慢日志支持根据时延进行过滤和排序，便于分析执行时长较久的慢日志。
3. 分布式集群支持查看节点倾斜，分片倾斜情况，便于分析性能是否均衡，保证资源的合理利用。
4. 支持查看数据库重启日志，便于分析数据库何时重启，是否发生故障等。

LightDB-EM 23.2 版本升级内容

1. 支持单机 HA 模式监控。

LightDB-EM 23.1 版本升级内容

1. 新增索引监控，查看数据库中的无效索引。
2. 新增顶级活动，可查看不同等待事件平均活跃会话，一定时间内 TOP SQL 占比，获取具体 SQL 内容，在线获取执行计划。
3. 新增动态修改 guc 参数，可通过页面修改 guc 参数信息。
4. 新增深度钻取数据库表年龄信息页面。
5. 新增角色，用户菜单，可通过添加角色，让不同用户拥有不同菜单权限。
6. 新增 sql 审核，可以系统审核 sql 文本，并且人工审核 sql 是否到指定的数据库实例执行。

LightDB-EM 22.4 版本升级内容

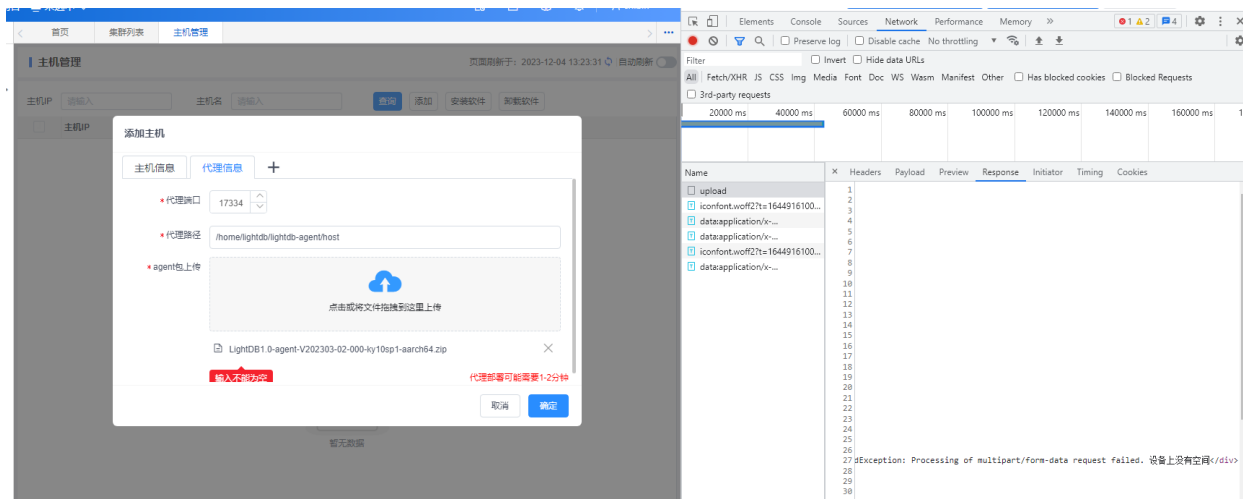
1. 支持配置邮件、钉钉告警。
2. 集成 lightdb 提供的最大可用性 checklist 及巡检项。
3. 内嵌支持绿色版 Lightdb。
4. EM 日志采集与检索。
5. 增加恢复后立即启动功能。
6. 支持 pwr 禁用 witness。
7. 高可用多从支持某些节点不提升。
8. 日志分析模块增加 SQL 审计。
9. 增加日志与进度。
10. 在集群列表部署高可用数据库实例。

LightDB-EM 22.3 版本升级内容

1. 主机监控和数据库实例监控分开部署；
2. 集群列表添加实例启停和主从切换，主机管理新增安装软件，配合集群管理可以初始化数据库实例；
3. 新增 top 对象，可查看数据库，数据库表，索引大小；
4. 采集监控添加下拉列表区分主机还是数据库实例的定时任务；
5. 新增备份恢复功能。

9 FAQ

1. 上传文件时，出现上传失败的错误



解决方法：

- a: 通过浏览器控制台查看上传接口提示:Processing of multipart/form-data request failed. 设备上没有空间
- b: em 配置文件 jrescloud.properties 配置 server.tomcat.basedir 的地址为 em 服务所在路径

2, em 内嵌的绿色版本的 lightdb 的版本 (13.8-22.4) 已支持单机分布式, 默认启动后监听端口为 5434, 如果手动修改了 em 内嵌的绿色版本的 lightdb 监听端口 (配置文件 \${LTDATA}/lightdb.conf 中配置项 port) 为 5436, 在 em 启动后 (em 启动过程中会启动绿色版本 lightdb 数据库) 需要进行如下操作:

```
1、[lightdb@192_168_76_128 scripts]$ ltsql -h localhost -p5436 -d postgres -c 'update_
↪cron.job set nodeport=5436;'
UPDATE 5
2、[lightdb@192_168_76_128 scripts]$ ltsql -h localhost -p5436 -d postgres -c "update_
↪servers set connstr='dbname=postgres port=5436';"
UPDATE 1
3、[lightdb@192_168_76_128 scripts]$ ltsql -h localhost -p5436 -d em -c "update pg_
↪dist_node set nodeport=5436;"
UPDATE 1
```

验证是否操作成功:

```
1、[lightdb@192_168_76_128 scripts]$ ltsql -h localhost -p5436 -d em -c "select *_
↪from pg_dist_node;"
nodeid | groupid | nodename | nodeport | noderack | hasmetadata | isactive | _
↪noderole | nodecluster | metadatasynced | shouldhaveshard
-----+-----+-----+-----+-----+-----+-----+-----
↪--+-----+-----+-----+-----+-----+-----+-----+-----
1 | 0 | localhost | 5436 | default | t | t | _
↪primary | default | t | t
(1 row)

2、[lightdb@192_168_76_128 scripts]$ ltsql -h localhost -p5436 -d postgres -c "select_
↪connstr from servers;"
connstr
-----
dbname=postgres port=5436
(1 row)

(1 row)

3、[lightdb@192_168_76_128 scripts]$ ltsql -h localhost -p5436 -d postgres -c "select_
↪* from cron.job;"
jobid | schedule | command | nodename | nodeport | _
↪database | username | active | jobname
-----+-----+-----+-----+-----+-----+-----+-----
↪-----+-----+-----+-----+-----+-----+-----+-----
1 | */5 * * * * | SELECT lt_catalog.pg_show_plans() | localhost | 5436_
↪| postgres | lightdb | t | lt_show_plans
2 | */1 * * * * | SELECT collect_activity_history() | localhost | 5436_
↪| postgres | lightdb | t | collect_activity_history
3 | */1 * * * * | SELECT collect_activity_profile() | localhost | 5436_
↪| postgres | lightdb | t | collect_activity_profile
4 | 0 0 * * * | SELECT clean_activity_profile() | localhost | 5436_
↪| postgres | lightdb | t | clean_activity_profile
5 | */10 * * * * | SELECT lt_catalog.take_sample() | localhost | 5436_
↪| postgres | lightdb | t | take_sample
(5 rows)
```