

# 管理系统-数据查询

## 1.录音记录查询

### 场景

- 查询范围(部门)内相关用户的录音记录
  - 管理员管理多个部门(可达上万个部门)
  - 每个部门对应多个用户
  - 每个用户存在多条录音

### 数据结构

- BMS数据库：管理员权限范围表

```
CREATE TABLE `bmsUserScope` (  
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `userId` varchar(64) CHARACTER SET utf8mb4 NOT NULL COMMENT '用户ID',  
  `tenementId` varchar(64) CHARACTER SET utf8mb4 NOT NULL COMMENT '企业ID',  
  `departmentId` varchar(64) CHARACTER SET utf8mb4 NOT NULL COMMENT '部门ID',  
  `descendantVisible` tinyint(3) unsigned NOT NULL DEFAULT '1' COMMENT '1表示包含子孙部门, 0表示不包含',  
  PRIMARY KEY (`id`),  
  KEY `userId` (`userId`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

- EIM数据库：部门用户表

```
CREATE TABLE `ucDepartmentUser` (  
  `departmentId` varchar(64) NOT NULL,  
  `username` varchar(64) NOT NULL,  
  PRIMARY KEY (`departmentId`, `username`),  
  KEY `ucDepartmentUser_username_idx` (`username` (20))  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

- Record录音记录数据库：录音记录表(按月分表)

```
CREATE TABLE `recordInfo_201904` (  
  `idx` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `cdrid` char(64) NOT NULL DEFAULT '0_0',  
  `caller` char(64) NOT NULL,  
  `callee` char(64) NOT NULL,  
  `callDir` int(11) NOT NULL DEFAULT '0',  
  `owner` char(64) NOT NULL,
```

```

`ownerType` tinyint(4) NOT NULL DEFAULT '0',
`ownerDepIDs` varchar(256) DEFAULT NULL,
`recName` varchar(256) NOT NULL,
`storageName` varchar(256) NOT NULL,
`startTime` int(8) NOT NULL,
`longTime` int(11) NOT NULL DEFAULT '0',
`fileSize` int(11) NOT NULL DEFAULT '0',
`TenantID` int(11) NOT NULL,
`isConnected` tinyint(1) NOT NULL DEFAULT '0',
`mark` tinyint(4) DEFAULT '0',
`expiredTime` int(11) NOT NULL,
`comment` varchar(256) DEFAULT NULL,
PRIMARY KEY (`idx`),
KEY `TenantID_idx` (`TenantID`),
KEY `cdrid_dir_idx` (`cdrid`,`callDir`),
KEY `Owner_timer_idx` (`owner`,`startTime`),
KEY `StartTime_idx` (`startTime`),
KEY `Caller_idx` (`caller`),
KEY `Callee_idx` (`callee`),
KEY `expiredTime_idx` (`expiredTime`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

## 查询语句

```

# 查询管理员可管理的部门列表 departmentIds
select departmentId from bms.bmsUserScope where userId = "u10001";

# 查询可管理的用户列表 userIds
select userId from eim.ucDepartmentUser where departmentId in (${departmentIds});

# 查询可见的录音记录 *${userIds}可能上万个*
SELECT
    idx, cdrid, caller, callee, callDir, owner, ownerType, recName, storageName, startTime,
    longTime, fileSize, TenantID as tenantId, isConnected, mark, comment
FROM
    Record.RecordInfo_201903
WHERE
    TenantID = 880001
    AND isConnected = 1
    AND startTime >= 1551369600
    AND startTime <= 1554047999
    AND (
        caller IN(${userIds})
        OR callee IN(${userIds})
    )

```

- explain

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	RecordInfo_201903	range	TenantID_idx,StartTime_idx	StartTime_idx	4	(Null)	2355128	Using where

## 2.挂机短信查询

### 场景

查询管理范围内相关“中继号码”关联的“短信内容”

- 每个管理员对应多部门(可达上万个部门)
- 每个部门关联多个中继号码
- 每个中继号码关联一条短信内容记录(如果有)

### 数据结构

#### 部门中继号码关联表

```
CREATE TABLE `bms_departmentOutwireNumber` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `tenementId` varchar(64) NOT NULL COMMENT '企业ID',
  `tenantId` int(10) unsigned NOT NULL COMMENT 'VOS企业ID',
  `departmentId` varchar(64) NOT NULL COMMENT '部门ID',
  `outwireNumber` varchar(32) NOT NULL COMMENT '中继号码',
  `trunkId` varchar(64) NOT NULL COMMENT '中继',
  `inputTime` bigint(13) unsigned NOT NULL DEFAULT '0' COMMENT '添加时间',
  PRIMARY KEY (`id`),
  KEY `departmentId` (`departmentId`),
  KEY `outwireNumber` (`outwireNumber`),
  KEY `trunkId` (`trunkId`) USING BTREE,
  KEY `tenantId` (`tenantId`)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=utf8 COMMENT='部门号码关系表';
```

#### 挂机短信记录表

```
CREATE TABLE `bms_onHookMessage` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `tenementId` varchar(64) NOT NULL COMMENT '企业ID',
  `tenantId` int(10) unsigned NOT NULL COMMENT 'VOS企业ID',
  `outwireNumber` varchar(32) NOT NULL COMMENT '中继号码',
  `trunkId` varchar(64) NOT NULL COMMENT '中继ID',
  `callInMsg` text NOT NULL COMMENT '呼入短信内容',
  `callOutMsg` text NOT NULL COMMENT '呼出短信内容',
  `callInEnabled` tinyint(1) unsigned NOT NULL DEFAULT '0' COMMENT '1开启 0 关闭',
  `callOutEnabled` tinyint(1) unsigned NOT NULL DEFAULT '0' COMMENT '1开启 0 关闭',
  `inputTime` bigint(13) unsigned NOT NULL DEFAULT '0' COMMENT '添加时间',
  `modifyTime` bigint(13) unsigned NOT NULL DEFAULT '0' COMMENT '修改时间',
  PRIMARY KEY (`id`),
  KEY `tenantId_outnum_trunkId` (`tenantId`, `trunkId`, `outwireNumber`) USING BTREE
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=utf8 COMMENT='挂机短信';
```

## 查询语句

```
# 查询列表 ${departmentIds} 可达上万个部门ID
# callInMsg,callOutMsg需要支持模糊搜索
SELECT
    a.tenantId,
    a.tenantId,
    a.departmentId,
    a.outwireNumber,
    a.trunkId,
    c.callInMsg,
    c.callOutMsg,
    c.callInEnabled,
    c.callOutEnabled,
    a.inputTime AS outwireInputTime,
    c.inputTime,
    c.modifyTime
FROM
    (
        SELECT
            *
        FROM
            bms_departmentOutwireNumber
        WHERE
            departmentId IN(${departmentIds})
    )a
INNER JOIN bms_onHookMessage c ON(
    c.tenantId = a.tenantId
    AND c.outwireNumber = a.outwireNumber
    AND c.trunkId = a.trunkId
    AND c.callInMsg LIKE CONCAT('%', '会计', '%')
    AND c.callOutMsg LIKE CONCAT('%', '98798', '%')
)
WHERE
    a.tenantId = '880001'
ORDER BY
    a.outwireNumber ASC
LIMIT 0,
15
```

- explain

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	PRIMARY	<derived2>	ref	<auto_key0>	<auto_key0>	4	const	30	Using temporary; Using filesort
1	PRIMARY	c	ALL	tenantId_outnum_trunkId	NULL	NULL	NULL	5	Using where; Using join buffer (Block Nested Loop)
2	DERIVED	bms_departmentOutwireNumber	ALL	departmentId	NULL	NULL	NULL	9667	Using where

## 3.来电彩铃查询

### 场景

查询管理范围相关“中继号码”的“彩铃配置”

- 每个管理员对应多个部门(可达上万个部门)

- 每个部门对应该多个中继号码
- 每个中继号码对就一条来电彩铃配置记录(如果有)
  - 彩铃从VOS的彩铃资源中选择

## 数据结构

### BMS彩铃配置表

```
CREATE TABLE `bms_callerColorBell` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `tenementId` varchar(64) NOT NULL COMMENT '企业ID',
  `tenantId` int(10) unsigned NOT NULL COMMENT 'VOS企业ID',
  `outwireNumber` varchar(32) NOT NULL COMMENT '号码',
  `trunkId` varchar(64) NOT NULL,
  `promptId` varchar(32) NOT NULL COMMENT '提示单(彩铃)ID',
  `inputTime` bigint(13) unsigned NOT NULL DEFAULT '0' COMMENT '导入时间',
  `modifyTime` bigint(13) unsigned NOT NULL DEFAULT '0' COMMENT '最后修改时间',
  PRIMARY KEY (`id`),
  KEY `outwireNumber` (`outwireNumber`),
  KEY `trunkId` (`trunkId`),
  KEY `promptId` (`promptId`)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=utf8mb4 COMMENT='来电彩铃';
```

### VOS彩铃文件资源表

```
CREATE TABLE `IvrPrompt` (
  `PromptID` char(128) NOT NULL COMMENT '保存文件名',
  `TenantID` int(11) NOT NULL DEFAULT '0',
  `OriginalFileName` char(128) NOT NULL COMMENT '原始文件名',
  `PromptName` varchar(128) CHARACTER SET utf8 COLLATE utf8_bin NOT NULL DEFAULT '' COMMENT '名称',
  `content` varchar(512) CHARACTER SET utf8 COLLATE utf8_bin NOT NULL DEFAULT '' COMMENT '语音内容',
  `LastModifyTimestamp` int(8) NOT NULL COMMENT '最后修改时间戳',
  `AuditResult` int(1) NOT NULL DEFAULT '0' COMMENT '审计结果, 0 未审计 1 未通过 2 通过',
  `RefCount` int(11) NOT NULL DEFAULT '0' COMMENT '引用计数',
  PRIMARY KEY (`PromptID`, `TenantID`),
  UNIQUE KEY `Name` (`TenantID`, `PromptName`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

## 查询语句

```
# 查询 ${departmentIds} 可达上万个部门
# PromptName,OriginalFileName,content需要支持模糊搜索
SELECT
  a.tenementId,
  a.tenantId,
  a.departmentId,
```

```

a.outwireNumber,
a.trunkId,
a.inputTime AS outwireInputTime,
d.OriginalFileName AS originFileName,
d.PromptName AS promptName,
d.content AS content,
d.AuditResult AS auditResult,
d.RefCount AS refCount,
d.promptId,
d.modifyTime
FROM
(
    SELECT
        *
    FROM
        bms_departmentOutwireNumber
    WHERE
        departmentId IN(${departmentIds})
)a
INNER JOIN(
    SELECT
        c.tenantId,
        c.outwireNumber,
        c.trunkId,
        e.OriginalFileName,
        e.PromptName,
        e.content,
        e.AuditResult,
        e.RefCount,
        c.promptId,
        c.modifyTime
    FROM
        bms_callerColorBell c
    INNER JOIN IvrPrompt e ON(
        e.PromptID = c.promptId
        AND e.PromptName LIKE "%" '897' %"
        AND e.OriginalFileName LIKE "%" '陈奕迅' %"
        AND e.content LIKE "%" '达' %"
    )
)d ON(
    d.tenantId = a.tenantId
    AND d.outwireNumber = a.outwireNumber
    AND d.trunkId = a.trunkId
)
WHERE
    a.tenantId = '880001'
ORDER BY
    a.outwireNumber ASC
LIMIT 0,
15

```

- explain

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	PRIMARY	<derived2>	ref	<auto_key0>	<auto_key0>	4	const	30	Using where
1	PRIMARY	<derived3>	ref	<auto_key0>	<auto_key0>	392	const,a.outwireNumber,a.trunkId	11	Using where
3	DERIVED	e	ALL	NULL	NULL	NULL	NULL	29	Using where
3	DERIVED	c	ref	promptId	promptId	130	func	1201	Using index condition
2	DERIVED	bms_departmentOutwireNumber	ALL	departmentId	NULL	NULL	NULL	9667	Using where