

# 数据库设计

## 一、数据库设计

数据库名称：dip

## 二、表设计

### 1、应用 (Application)

表名称：t\_application

字段英文名	字段中文名	数据类型	是否允许为空	索引	备注
id	主键ID	BIGINT	否	主键	自增主键
key	应用唯一标识	CHAR(32)	否	索引	
name	应用名称	VARCHAR(128)	否	--	
description	应用描述	VARCHAR(800)	是	--	
icon	应用图标	Blob	是	--	应用的图标
category	应用所属分组	VARCHAR(128)	是	--	
version	应用版本号	VARCHAR(128)	是	--	version 必须符合语义版本号。
micro_app_config	微应用配置	JSON	否	--	前端微应用加载信息 Example: <div style="border: 1px solid #ccc; padding: 10px; width: fit-content;">代码块 1 {</div>

```

2     "name": "intelligent_fault_analysis",
3     "entry": "/intelligent_fault_analysis",
4     "headless": false
5   }

```

release_config	应用安装配置	JSON	否	--	记录应用安装了哪些helm release Example:["release1","release2"]
ontology_config	业务知识网络配置	JSON	是	--	多个业务知识网络配置 Example:  代码块 <pre>1 [ { "id": 201, "is_config": true }, { "id": 202, "is_config": false } ]</pre>
agent_config	智能体配置	JSON	是	--	多个智能体配置 Example:  代码块 <pre>1 [ { "id": 201, "is_config": true }, { "id": 202, "is_config": false } ]</pre>
updated_by	更新者 ID	CHAR(36)	否	索引	更新应用包的用户的 ID
updated_at	更新时间	TIMESTAMP	否	--	

## 2、用户 (User)

表名称: t\_user

	字段中文名	数据类型		索引	备注

字段英文名			是否允许为空		
id	主键 ID	BIGINT	否	主键	自增主键
user_id	用户 ID	CHAR(36)	否	索引	
display_name	用户显示名称	VARCHAR(255)	否	--	

### 3、角色 (Role)

表名称：t\_role

字段英文名	字段中文名	数据类型	是否允许为空	索引	备注
id	主键 ID	BIGINT	否	主键	自增主键
role_id	角色 ID	CHAR(36)	否	索引	
role_name	用户显示名称	VARCHAR(255)	否	--	

系统内置固定角色：

- 普通用户：49c9f0a6-6f3b-4f4f-9f3b-1c6a7d9d3c12
- 应用管理员：1b4f9c46-9c3c-4c36-8e3e-7df3e6c2f7ad

### 4、用户-角色关系 (User\_Role)

表名称：t\_user\_role

字段英文名	字段中文名	数据类型	是否允许为空	索引	备注
user_id	用户 ID	CHAR(36)	否	联合主键	关联 t_user.user_id
role_id	角色 ID	CHAR(36)	否	联合主键	关联 t_role.role_id

## 二、实体关系

视图 | ?

```
1 erDiagram
2   Application {
3     string key PK
4     string updated_by FK
5   }
6
7   User {
8     string user_id PK
9   }
10
11  Role {
12    string role_id PK
13  }
14
15  User_Role {
16    string user_id PK
17    string role_id PK
18  }
19
20  User ||--o{ Application: "user_id → u"
21  Role ||--o{ User_Role: ""
22  User ||--o{ User_Role: ""
23
```

