用户users表

|  |  |
| --- | --- |
| id |  |
| username |  |
| password |  |
| salt |  |
| locked |  |
| create\_time |  |
| last\_update\_time |  |
| create\_by | 创建者 |
| update\_by | 更新者 |
| department\_id | 部门id |
| job\_id | 岗位id |
| nick\_name | 用户昵称 |
| gender | 性别 |
| phone | 手机号码 |
| email | 邮箱 |

部门depts表

|  |  |
| --- | --- |
| id |  |
| parent\_id | 上级部门id |
| level | 部门层级 |
| name | 名称 |
| create\_time | 创建时间 |
| last\_update\_time | 最后更新时间 |

岗位jobs表

|  |  |
| --- | --- |
| id |  |
| name | 岗位名称 |
| enabled | 岗位状态 |
| dept\_id | 所在部门id |
| create\_time | 创建日期 |
| last\_update\_time | 最后更新时间 |

角色roles表

|  |  |
| --- | --- |
| id |  |
| role |  |
| desc |  |
| create\_time |  |
| last\_update\_time |  |
| level | 角色级别 |

用户角色users\_roles表

|  |  |
| --- | --- |
| id |  |
| users\_id | 用户id |
| roles\_id | 角色id |

部门角色depts\_roles表（当前角色能够管理的部门）

|  |  |
| --- | --- |
| id |  |
| roles\_id | 角色id |
| depts\_id | 部门id |

可以为角色自定义可管理的部门，一个用户可以有多个角色，一个角色可以管理多个部门（包括一个部门的所有子部门）。

角色权限roles\_permissions表（当前角色对应的增删改查权限）

角色与权限属于多对多关系；

|  |  |
| --- | --- |
| id |  |
| role\_id | 角色id |
| permission\_id | 权限id |

可以为角色自定义操作权限，一个用户可以有多个角色，一个角色可以有多个权限。

permissions表

|  |  |
| --- | --- |
| id |  |
| permission | 权限 |
| description | 描述 |
| create\_time | 创建时间 |
| last\_update\_time | 最后更新时间 |

一个用户有一个角色，可以自定义每一个角色管理的部门，自定义角色的操作权限；

一个用户有多个角色，每一个角色有自己要管理的部门，有自定义的操作权限。

当我想让一个部门的管理员同时管理另一个部门时:

* 用户只能有一个角色，就给他换一个角色：不一定有满足要求的角色（同时能够管理这两个部门的角色），如果没有就得去创建，就会产生许多冗余角色，不简洁。
* 用户可以有多个角色，给他添加一个角色：添加另一个部门的管理员的角色。

所以一个用户还是有多个角色比较好。