3.区域权限

3.1 基本权限类型

1. 不能进入。
2. 不能出去。
3. 不能呆太久。
4. 不能出去太久。
5. 不能靠近边界。（围栏功能）

3.2 使用场景

1. 告警围栏

不具有权限的人都不能进入，进入则发送告警。

具体场景：制氢站不能进去，废水处理间不能进去，危险化学物品房间不能进去，发电机或者某一个大型设备范围不能靠近。

告警围栏也可能分等级。

根据靠近的程度，可以先发送轻微提示；具体靠近到一定程度后，发送警告，要求离开。

1. 工作区域划定

在一定时间段中必须在某一个区域内；或者多个区域内，区域之间移动时可以有一定的离开时间。离开该区域（超过误差距离，超过误差时间）时，提示告警。

具体：不能离开工作场所太久；执行检修某些设备时必须在设备所在的区域。

1. 限制区域进入时间

从进入开始，在某一个时间范围内不离开该区域，则告警。

具体：休息区、吸烟区等不能停留太久。

1. 巡检人员监控

人员必须在某一个时间范围到达一个地点，在一个时间范围内不能从一个区域出去。进去后就不能出去。

创建巡检路线区域，巡检区域告警：在某一个时间内，巡检任务时间，离开了巡检区域。

巡检路线关键点告警：没有在一个时间点+误差范围内，到达某一个关键点。

1. 危险物资（人物，如监狱环境下）监视

从进入开始，永远不能（直到移除了这个规则）离开某一个区域，离开便告警。

有些场景可以不用发告警，但是要记录下可疑行为，做后续的数据分析。

对于定位引导这种功能，和巡检有点像，从一个点到下一个点，指引人移动，感觉需要移动端App的支持；告警也是，短信、邮箱、电话以外直接发送App消息推送也是一个方式。（不用在意这个，随便想想的）

3.3 创建权限（AreaAuthority,区域权限）

1. 规则组（AccessRuleGroup）

整理规则以及扩展用，可以一开始有一个默认的组，所有的新增规则默认都数据该组。

属性:Id,Name,Describe,Pid(父的组，默认为空),Rules(规则列表)

一个规则只能属于某一个规则组。

界面操作相关：可以在创建权限时选择一个组，就添加了组内的所有规则。

规则组没有实际作用，仅仅是为了操作上方便设想出来用于分组和批量操作规则用的。

1. 进出规则（AccessRule）

设计思路：

时间段

开始时间(StartTime)、结束时间(EndTime)

时间长度(TimeSpan)

时间长度，进入时间作为开始时间，计算出开始结束，超过结束时间则告警。

时间重复

是否重复，一次性的，每天，每周（周一.周二.周三.周四）

时间误差范围

开始时间-开始误差时间(StartErrorRange)

结束时间+结束误差时间(EndErrorRange)

进出限制类型：

不能进入，

不能出去，

可以进入，

可以出去：在某一个时间范围内可以出去，除此之外，必须在区域内。

不能呆太久：在某一个时间范围内可以进入，除此之外不能进入。

属性信息:

基本：

Id,Name,Describe,

时间：

TimeType:时间设定方式，

1. 根据时间长度，只能呆一段。这种状态只能设置TimeSpan，不能设置StartTime和EndTime。
2. 根据提前设定的时间点，从开始时间到结束时间。

StartTime,EndTime,TimeSpan,StartErrorRange,EndErrorRange

重复：

AccessType:NoEnter,NoLeave,CanEnter,CanLeave,CanStay,NoStay,

RepeatType:Day（每天）,Once（只一次）,WorkDay（周一到周五），Custom（自定义）,Temporary(临时的)

CustomDay:自定义的星期几，具体保存格式待定：1\_2\_3\_4\_5\_6\_7，1:0:1:0:1:0:0（某一个位置是1，代表有这一天），0000000。

其他:

**Areas:关联区域列表，列表结构**。区域和规则是多对多的关系，不过操作上是创建规则时选择区域。

AreaErrorLength:区域误差距离。复杂一些的话，每个区域都有不同的误差距离。

Enable:是否激活

Tags:标签，分类用

GroupID:规则组，整理用

修改记录：

CreateTime:创建时间

ModifyTime:修改时间

DeleteTime:删除时间

关于时间段：可能需要可预定义一些时间点，让操作上比较方便，如：上午上班时间，下午上班时间，白天上班时间（上午+下午），XXX部门上网上班时间。

后续操作友好性优化可以考虑做起来。

1. 权限组（AreaAuthorityGroup）

定义：多个不同的权限的集合，分组用。

一个权限只能是一个分组，默认一个权限组。

属性:Id,Name,Describe,Pid(父的组，默认为空),Authoritys(权限列表)

界面操作相关：

1. 绑定：绑定权限时可以选择一个权限组就绑定了组内的全部权限。
2. 自动生成描述：最好能够根据规则的内容自动生成一段描述，如“根据规则1，可以进入区域1；根据规则2，不能进入区域2”。

权限组没有实际作用，仅仅是为了操作上方便设想出来用于分组和批量操作权限用的。

1. 权限（AreaAuthority）

定义：多个不同的区域进出规则的集合。

属性:Id,Name,Describe,GroupID(组),Authoritys(权限列表)

一个权限可以属于多个不同的规则，一个规则可以属于不同的权限。

最终用于绑定的是权限。

如：

权限1：规则1(可以进入区域1)+规则2(不能进入区域2)+规则3(不能离开区域3)

权限2：规则1(可以进入区域1)+规则4(可以进入区域2)

判断时，根据权限获取规则列表，判断规则，发送告警。

可能问题1：不同规则间相互矛盾，如：

规则A可以进入区域A，规则B不能进入区域A。

处理方案1：

1. 智能判断并限制用户选择。最理想是能够判断出来相互矛盾的规则并指出来，不让同时包含在一个权限中。规则中有个时间段的概念在里面，要判断的话要在同一个时间段中，对于同一个区域，有相互矛盾的规定。时间段可能重叠，区域也可能重叠，比较复杂。
2. **不用管**。因为告警的判断是相互独立的，可以在互相不影响的情况下发送自己规则的告警。
3. 时间长度使用方式

应用场景：不能在一个区域内呆超过5分钟

方案：在人员进入一个区域时，根据时间长度和进入时间点，创建一条临时权限（有开始时间和结束时间），绑定到人员上，人员的权限用这个创建的权限进行判断。结束时间到期后，将该权限删除（设置删除时间）。

3.4 绑定权限

1. 权限是绑定到人上还是绑定到卡上？

与人相关的可以是角色、岗位、部门等，与卡相关的是卡组。绑定到卡上，将卡发给某个人，某个人就具有了该卡的权限。卡上要标示出其编号，卡组，作用。绑定到人上，将卡发给某个人，该卡就具有了该人的权限。卡仅仅用于定位用，权限无关。

绑定到人员上的话，相关的东西比较多，设计起来比较复杂，另外应该说最终绑到人上，还是到卡上都能实现相同的效果，哪个简单哪个来。

另外，定位标签卡可以多人重复使用的。

**决定绑定到卡上吧**。

2. 直接绑定到卡上，还是卡组上(TagGroup)，或者是卡角色上(TagRole)？

感觉事先配置的工作和后续功能调整的工作都会挺多的。

为了应对改变，要解耦合，解耦合的基本方式是增加层次。

还是**绑定到卡角色上**吧，人员->卡->卡角色->权限->规则，后续可能改动是卡角色和人员直接绑定，而不是通过卡。

下面的功能相当于对卡管理的扩充。

卡角色(TagRole)，

一个卡有多个角色，每个角色有一个权限，也就是一个卡可以有多个权限，权限之间相互矛盾没关系，判断权限并发送告警相互不干扰。

属性:Id,Name,Describe,(父的组，默认为空),Tags(标签列表)

角色再往上分组的话，就是“域”的概念呢了，先不弄。

卡组(TagGroup)

分组用，一个卡只能属于某一个组。

属性:Id,Name,Describe,Pid(父的组，默认为空),Tags(标签列表)

卡（Tag）表要做相应修改

属性增加:Roles（TagRole列表）,Group（TagGroup）

如何实现：一次将一个权限赋予一批人（一个班组：1号路线巡逻组）？

通过界面操作的方式，通过表格筛选的方式选择关联到某一个班组人员的所有卡，关联某一个卡角色。

3.5 发卡：人员和标签卡绑定

先录入所有的标签信息，然后录入人员信息。

人员信息和标签卡信息都是独立录入的，不过在录入人员时可以选择关联哪个卡（~~理论和实际操作上都是可以有多个卡的，一个人口袋里放着多个标签卡嘛~~），也可以先不关联卡。

人员关联卡的操作机位发卡，发卡后也要激活卡才能显示到三维中并进行规则判断。

~~一个人员可以有多个卡，一个卡只能有一个人员。~~

人员和卡的绑定关系是1对1的。

一个卡可以给多个人使用，一个人可以使用不同的卡。

位置轨迹的记录信息要包括卡绑定的人员的信息，需要修改历史位置信息表。

查询历史轨迹时，可以按照卡查找，也可以按照人查找。

甚至可以查找某一个部门的所有人员的轨迹，同时显示多个轨迹，用不同的颜色（或者其他效果）区分不同的轨迹。

三维环节中的基本信息要有人员名称和卡名称。

3.6 区域告警

标签卡的告警类型：区域告警/消失告警/低电告警/传感器告警/重启告警/非法拆卸告警。

**现在只有区域告警，没有其他告警**。

**1.触发告警**

只有发卡后标签的位置信息才算是有效，所有三维中界面上是人员列表，没有标签列表。根据人员获取权限规则，判断当前位置是否符合权限规则，不符合则发出告警。

人员->卡(1:1)->卡角色(1:n)->权限(1:1)->规则(1:m)，一个人员可以有n\*m条规则。每一条规则独立判断，不相互影响。

每次实时位置信息发送过来，都要判断一下，是否符合规则，不符合时则发送告警。

是否告警也要存入历史数据中，可以在三维中区分出告警和没有告警的轨迹。（暂时新不动，具体怎么修改历史位置信息表，后续细化。）

考虑到性能问题，是否需要直接缓存人员和规则的关系，不用每次都到数据库中查询。

判断逻辑：

1.时间，当前时间点是否是规则规定的时间范围内

允许进入的时间，允许出来的时间。

2.区域，当前位置和规则中的区域的距离

是否在区域内，距离是多少

3.限制行为类型。

**2.告警列表**

告警信息：类型/等级/规则/终端/目标/时间/处理人/处理类型/处理时间/告警内容

告警列表界面：

列：类型/等级/规则/终端/目标/时间/处理人/处理类型/处理时间/告警内容/告警处理（该列界面上有操作按钮：处理/误报/忽略）

**3.告警处理**

界面上点击处理按钮，记录处理信息，有需要的话可以加一个处理详情。

**4.告警通知**

1.监控人员查看。

1.告警列表

2.告警弹窗。右边弹出，不断积累，不断消失。

2.监控人员通知。

邮件、短信通知，要设置一个全局管理人员，或者每个区域。

3.监控目标通知

短信、邮件、**电话**、App通知。

**5.疑问解答**

1. 告警现在还有没有告警等级划分的内容？还是后面就按照设想的标签卡的告警类型来？

要告警等级，用“一级、二级、三级、四级，分别用红色、橙色、黄色、蓝色标示，一级为最高级别”，设想是定位告警按违反规则的时间和距离来区分告警等级，进入一个不被允许进入的区域1分钟和10分钟是不一样的，超过10分钟就需要有人过去查看了。

1. 告警列表里面的终端、目标、处理类型是什么？

终端就是定位标签，目标就是绑定的人员（或者物资），处理类型有处理、忽略、误报。

1. 处理告警的方式感觉需要确定一下有什么内容？如果只有通知这种简单层级的话，感觉系统自动记录就可以了，好像不怎么需要手动记录（但是功能保留用于特殊情况）；不然感觉需要在人员定位的告警上面，给一个“定位到告警列表”的功能——就是我在不是列表的地方进行处理了，而我又需要手动记录，需要有一个立即跳转让我去记录的地方。

告警处理里不需要记录内容，当前的作用就是表面这个告警，监控人员注意到了，并处理了，具体怎么处理（通知人员还是其他），不用记录。