**库位分配算法，可测试版交付——7、8月份**

1. **提交一个可试运行原型，7月底最好**
2. **会以具体要交付的实际项目为基础**
3. **随机生成库位请求**
4. **根据测试结果进行完善**
5. **尽可能规避项目结构上的更改，初期结构设计上的考量**
6. **仅考虑Model-Service层以及往后的层**
7. **4月10号左右外部一人做UI介入**

Client一体。

**通用性算法 <--> 是否可以预留一个可扩展接口，一边在原通用性算法上，进行自定义的扩展？ <--> 策略扩展？<--> 管道接口 + 策略优先级 <--> 约定接口，可以对接口进行实现（抽象类），**

4月1日——7、8月份 :: 期望

🡪4月15日提供实际项目细节

🡪4月30日数据库、数据结构设计确定

🡪5月底之前完成算法和架构设计

🡪6、7月份开发

🡪7月份开始内部的开发测试、迭代

🡪8月交付初始成果，对方测试

1. Good数据表，增加GoodId和托盘条码（全为nchar）
2. MapItem与Good表的一对多依赖
3. Zone表的抽离，基础型写死在程序，区域存储数据库，扩展颜色属性；默认为货位（待做）
4. 暂时单品规、批次、多托货物模拟，考虑以下约束。

约束选项：

约束选项提供在一个界面中进行全局设置，计算时使用这些全局变量

1. 不同品规、批次混放；**品规和批次是联合主键；尽量算法一致**
2. 如果一个货道两端都有轨道，则分配时时优先顶到最深处还是放到中间
3. （暂时）设置一个换层偏好选项
4. 货道编号----输出的对应关系，可扩展接口；；给货道编号；；约定行方向along为轨道；列方向along为货道
5. 封装一个Order类，（订单拆分接口预留）；dest提供默认值
6. 外围加一圈，显示策略：默认障碍物或者不提供视图
7. 接口更变：一托货物和一订单货物，统一：获知货道：：货位这两个接口，锁货道数量而不是位置，具体分配货位等货物到达货道再分配
8. **对应数据库数据结构设计**