

【124智慧物料采购平台】开发文档v1.0

平台逻辑

1. **初始化地图阶段：** 将所有平台状态信息读入 对于**复合生产型工作台** 我们会生成对应的**生产任务** 加入到任务队列 然后修改平台的标识位 **【是否发布任务】** 为true 防止重复发布任务 **【同时需要在Task数据结构中存上任务对应的平台】**

难点一：如何更新**是否发布任务**这个标志位？ 目前想法，平台存有需要的物品类型，还有当前拥有的物品，比如need=0b111000

status=0b111000， 就代表此刻已经收集到了所有物品，那么下一帧 这个工作台就能可能投入生产 为什么是可能？ 因为有可能平台正在生产东西 所以还需要一个判断条件 就是剩余生产帧数为0

因此，总的条件应该是(((need >> 1) & (status >> 1)) == (status >> 1) && leftTime == 0)

2. **每一帧的输入数据处理：更新平台信息：** 若平台的 **【是否分布任务标志位】** 为false，则生成任务，加入任务队列；若平台的**产品格为1** **【可以根据status的第0位二进制位获得】**，则需要发布收原材料的任务，以便让机器人来收这个成品。

难点一：如何防止多个机器人来收这个成品，同样的，设置 **【是否发布fetch任务标志位】**，发布了任务就设置为true。更新这个标志位则是在每一帧的**平台信息status**中即可，若发现产品格为0，则重新设置为false。

所以当产品格为1的时候，是否需要发布fetch任务，只需要判断(status & 1) == 1 && 标志位 == 0

任务队列逻辑

1. **优先级问题**

总体逻辑：生产7号材料优于4，5，6号材料，fetch7，4，5，6号材料高于一切生产任务

此外，生产复合材料分解出来的子任务优先级应该更高。

这是为了防止子任务被后面高优先级的复合任务挤下去。

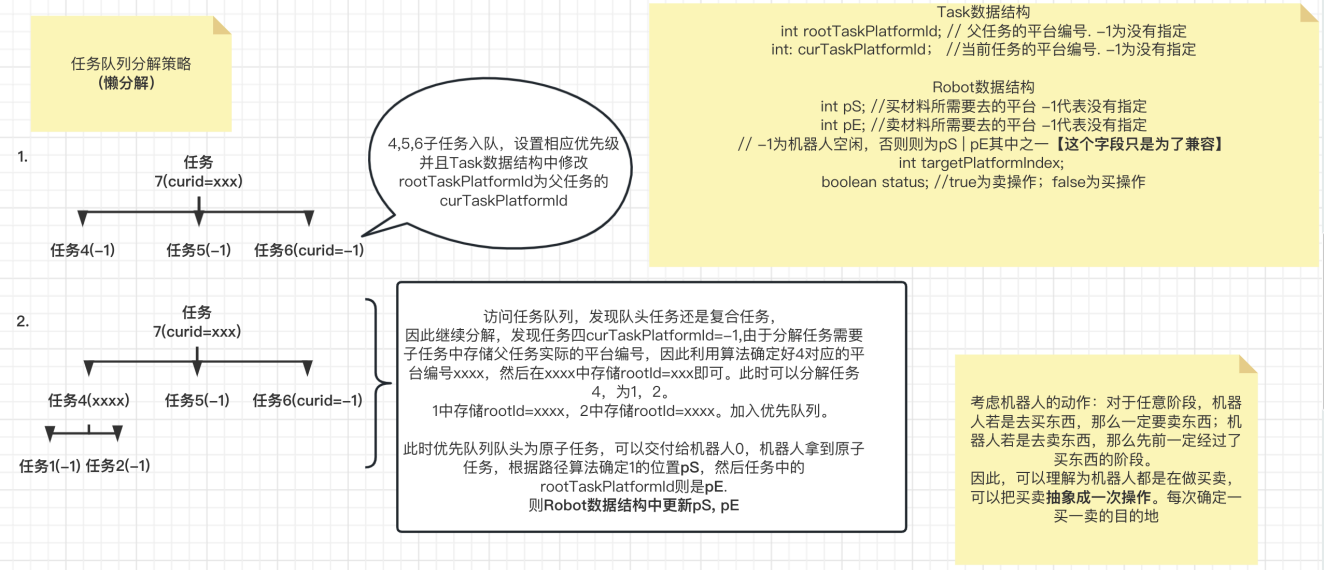
例如：队头[7,7,...]队尾， 将7分解成4,5,6，若4，5，6优先级比7低，则一个7出队后，优先队列变成

队头[7,4,5,6...]。这样以来，子任务的分解就没有意义了。

当在生产队列中遇到复合任务时，我们需要分解任务，例如现在队列头 **【7,4,5,6,...】** 队列尾，**当检测到机器人空闲**，就会访问这个优先队列的队头，取出7，发现7为复合任务，于是**分解任务7，为4，5，6**，**【由于这种分解出来的4，5，6实际上是为特定的7服务的，因此4，5，6的优先级应该更高于7，比如调用函数splitTask(rootTask, priority=7.priority - 1)】**，这个分解出来的4，5，6是**Task数据结构**，由于4，5，6是要送往特定的7，因此我们需要在Task中存入 **【指定的rootplatformId】**，然后在确定在哪个平台生产4或5或6时，再在对应的平台中修改其 **【rootPlatformId】** 为 **【指定的rootTaskPlatformId】**

2. 任务分解具体细节

采用【懒分解策略】，比如初始任务队列为队头【7,4,5,6...】，对于这些复合任务，当机器人真正有空去领取任务的时候，才进行分解，减少每帧的处理任务。



机器人视角

每帧迭代4个可用的机器人

1.若机器人空闲【待分配】

查找任务队列

队头取任务，若为复合任务，则调用分解任务函数。

取到的任务类型：都是买卖。区别在于

- 任务没有指定卖材料的目的地：比如生产7号材料平台，待7号材料生产出来的时候，平台会发布一个fetch任务，这类型的任务就没有目的地，机器人取到此类任务，利用算法规划一个卖材料的目的地即可。
- 任务指定卖材料的目的地：比如生产7号材料分解出来的4号任务，该4号任务的数据结构Task内就存储了rootTaskPlatformId，即卖材料的目的地。

若任务队列为空

说明该地图没有复合型生产工作台。那么只需要考虑买1，2，3 往9号类型平台输送即可。

2.若机器人繁忙【可能需要更新相关数据结构】

数据结构

Platform数据结构至少需要的成员

```
boolean: isAssignProductTask 是否发布生产任务
boolean: isAssignFetchTask 是否发布取的任务
int: rootPlatformId 生产的产品应该送往哪个平台【-1 表示没有指定 [0, n - 1]表示指定平台】
```

Task数据结构至少需要的成员

```
int: rootTaskPlatformId 父任务对应的平台，一般用于指定生产出来的产品该送去哪个平台进行进一步加工【-1表示没有】  
int: curTaskPlatformId 当前任务对应的平台 【-1表示尚未指定】
```

Robot数据结构至少需要的成员

```
int: pS; //买材料需要到达的平台编号 -1为没有指定  
int: pE; //卖材料需要到达的平台编号 -1为没有指定  
int: targetPlatformIndex; //目标平台编号 -1表示机器人空闲 其取值可能是pS或者pE 【此字段仅为了兼容之前版本】  
boolean: isSellStatus; //表示当前机器人的目的 true为卖操作 false为买操作
```