### 4.1.2Transportbl模块

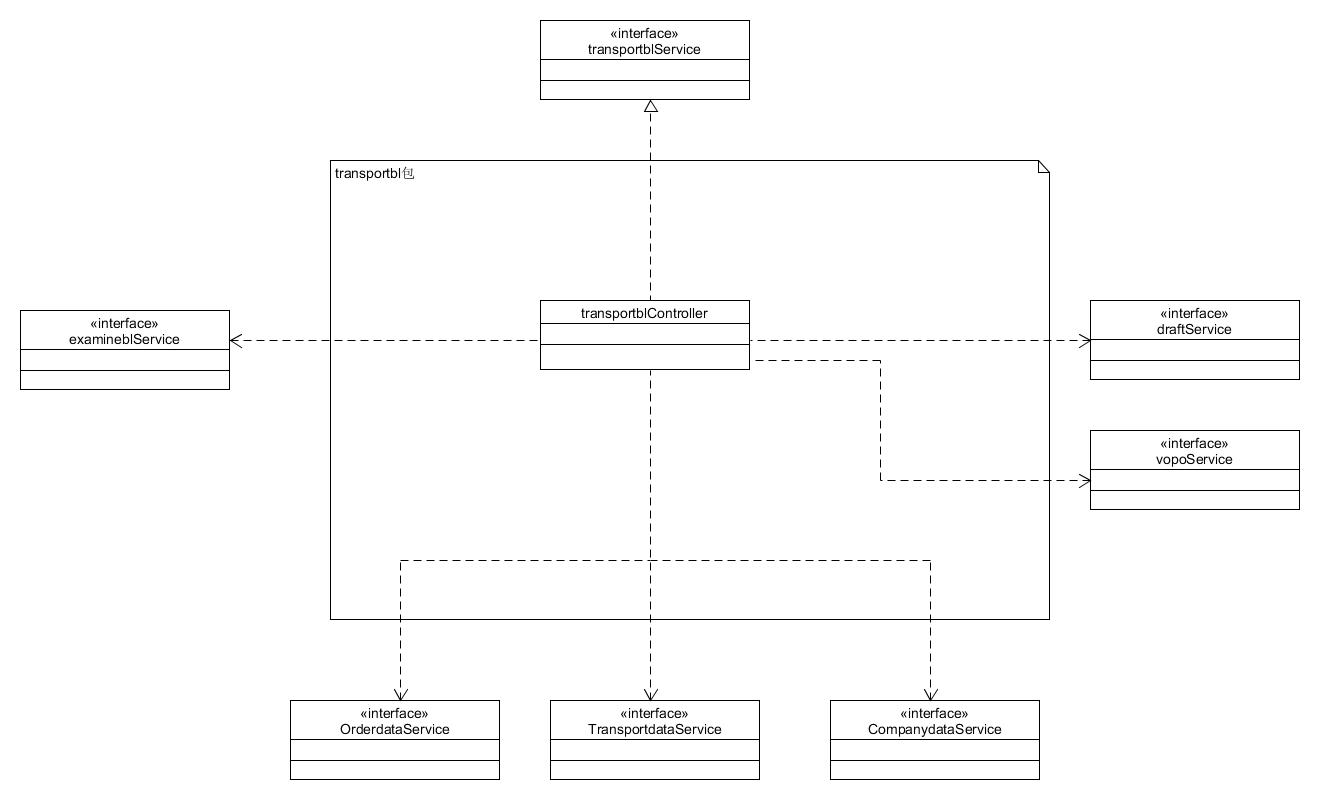
1. 模块概述

Transportbl模块负责实现营业厅装车单和中转中心中转单的新建、提交工作。

1. 整体结构

事实上，个人感觉这个包被架空了。一方面，包本身的逻辑比较简单；同时，它所需要实现的功能与其他新建订单包类似，因此采用了统一的工具实现。因此包内部的组织结构就相对较为简单

Transportbl模块的设计如图



Transportbl模块各个类的职责

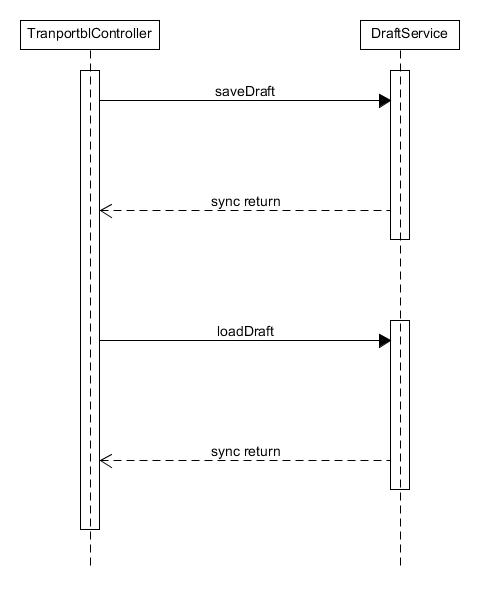
|  |  |
| --- | --- |
| 模块 | 职责 |
| TransportblController | 负责实现中转/装车单的服务 |

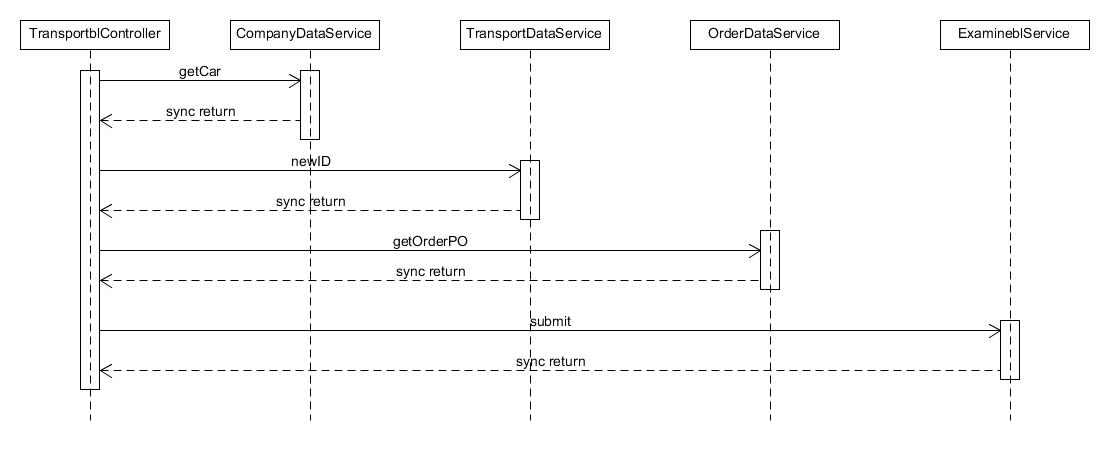
1. 模块内部类的接口规范

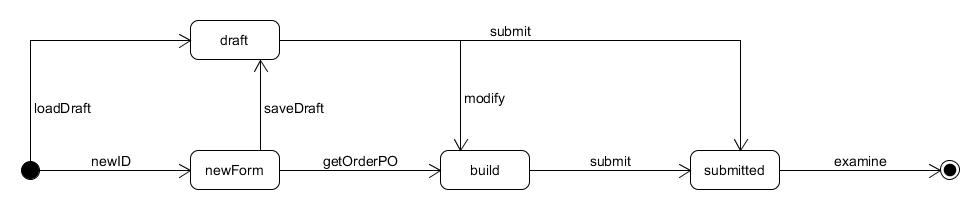
**TransportBL的接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| TransportblController. checkFormat | 语法 | public List< CheckFormMessage > checkFormat(LoadPO form, boolean isFinal) |
| 前置条件 | 输入的LoadPO不为空,isFinal判断是否为提交表格的操作 |
| 后置条件 | 检查输入信息是否符合系统要求，若符合，返回的list中只有一个显示正确的CheckFormMessage, 否则返回能指示出错误的CheckFormMessage的列表 |
| TransportblController.newID | 语法 | public String newID() |
| 前置条件 | 无 |
| 后置条件 | 返回一个新的装车单编号 |
| TransportblController.getCar | 语法 | public CarVO getCar(String hallID) |
| 前置条件 | 传入营业厅编号 |
| 后置条件 | 返回可用车辆的信息 |
| TransportblController.submit | 语法 | public OperationMessage submit(LoadPO form) |
| 前置条件 | LoadPO已经通过检查 |
| 后置条件 | 若网络连接正常，返回正常的OperationMessage，否则返回相应的提示信息 |
| TransportblController.saveDraft | 语法 | public OperationMessage saveDraft(LoadPO draft) |
| 前置条件 | 输入的LoadPO不为空 |
| 后置条件 | 将LoadPO序列化成文件存储 |
| TransportblController.loadDraft | 语法 | public LoadPO loadDraft() |
| 前置条件 | 有对应的序列化文件存在 |
| 后置条件 | 将序列化文件解序列化成对应的LoadPO对象返回 |
| TransportblController. checkFormat | 语法 | public List<OperationMessage> checkFormat(CenterPO form) |
| 前置条件 | 输入的CenterPO不为空 |
| 后置条件 | 检查输入信息是否符合系统要求，若符合，返回的list中只有一个显示正确的OperationMessage, 否则返回能指示出错误的OperationMessage的列表 |
| TransportblService.newID | 语法 | public String newID() |
| 前置条件 | 无 |
| 后置条件 | 返回合法的新中转单编号 |
| TransportblController.submit | 语法 | public OperationMessage submit(CenterPO form) |
| 前置条件 | CenterPO已经通过检查 |
| 后置条件 | 若网络连接正常，返回正常的OperationMessage，否则返回相应的提示信息 |
| TransportblController.saveDraft | 语法 | public OperationMessage saveDraft(CenterPO draft) |
| 前置条件 | 输入的CenterPO不为空 |
| 后置条件 | 将CenterPO序列化成文件存储 |
| TransportblController.loadDraft | 语法 | public CenterPO loadDraft() |
| 前置条件 | 有对应的序列化文件存在 |
| 后置条件 | 将序列化文件解序列化成对应的CenterPO对象返回 |
| 需要的服务（需接口） | | |
| 服务名 | 服务 | |
| ExamineService.  submit(FormPO po) | 提交表格审批 | |
| TransportDataService.newID | 得到新的中转单/装车单号 | |
| OrderDataService.getOrderPO | 获得订单信息 | |
| CompanyDataService.getCar | 获得可用车辆信息 | |

1. 业务逻辑层的动态模型







1. 业务逻辑层的设计原理

采用集中式的控制