1. 模块概述

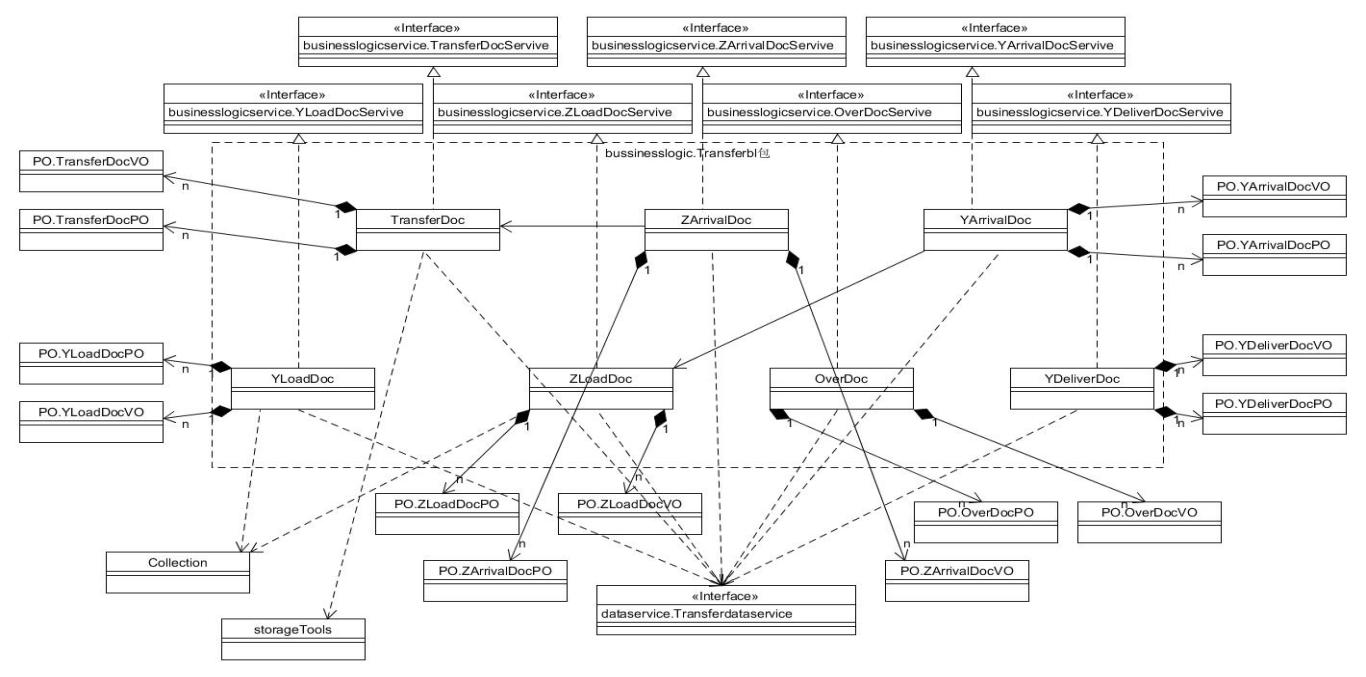
Transferbl模块承担的需求参见需求规格说明文档功能需求及相关非功能需求

Transferbl模块的职责及接口参见软件体系结构描述文档

1. 整体结构

展示层、业务逻辑层、数据层间由TransferLogicService接口与TransferDataService分隔，业务逻辑根据不同操作对象，由TransferDoc、ZArrivalDoc等对象完成，TransferDocPO、ZArrivalDocPO等是各单据的持久化对象，TransferDocVO、ZArrivalVO等是各单据的可视化对象。

Transferbl模块设计如图所示



1. 模块内部类的接口规范

TransferDoc的接口规范

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 供接口 | | | | | |
| Transferbl.getUncheckedTransferDoc | 语法 | | Public TransferDocPO getAllTransferDoc() |
| 前置条件 | | 无 |
| 后置条件 | | 得到待审批中转单 |
|  | | | | | |
| 需要的服务 | | | | |
| 服务名 | | 服务 | | |
| CollectionDataService.saveSendDocPO(SendDocPO po) | | 保存寄件单单一持久化对象 | | |
| CollectionDataService.getDistance(String city) | | 获取距离数据 | | |
| CollectionDataService.getSequence() | | 获取序列号 | | |
| CollectionDataService.changeSequence(int sequence) | | 改变最后序列号 | | |
| CollectionDataService.saveHistory(HistoryPO po) | | 保存历史数据单一持久化对象 | | |

YDeliverDoc的接口规范

OverDoc的接口规范

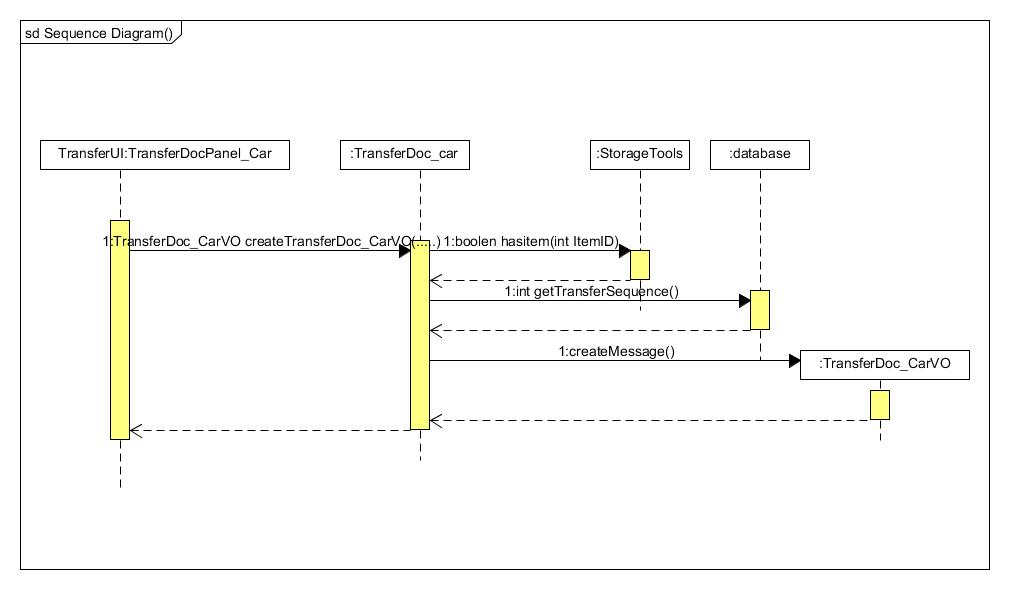
YLoadDoc的接口规范

ZLoadDoc的接口规范

YArrivalDoc的接口规范

ZArrivalDoc的接口规范

1. 业务逻辑层的动态模型

图示表明了系统中，中转中心业务员选择汽运中转，输入到达地，车次号，监装员和所有单号后，揽件逻辑处理的相关对象之间的协作

1. 业务逻辑层的设计原理

利用分散式控制风格，每个操作不同对象的业务逻辑由不同的逻辑对象提供