体系结构设计变更记录

1. FinanceBl模块的接口规范
2. 变更原因：

FinanceBl模块集合了成本管理、结算管理、审批单据、制定价格及查看统计报表任务，太过臃肿，为了减少模块间的耦合，我们在不改变FinanceBl模块职责的前提下，将FinanceBl模块分成五个子模块，分别对应上述五个任务。

1. 变更前

|  |  |  |
| --- | --- | --- |
| **提供的接口（供接口）** | | |
| Finance.approval | 语法 | public boolean approval(DocumentVO document) |
| 前置条件 | 启动一个审批单据回合 |
| 后置条件 | 在一个审批单据回合中，判定单据是否通过 |
| Finance.getPrice | 语法 | public double getPrice(Express type) |
| 前置条件 | 启动一个制定价格回合 |
| 后置条件 | 在一个制定价格回合中，返回指定类型的快递价格 |
| Finance.pricing | 语法 | public void pricing(double price) |
| 前置条件 | 启动一个制定价格回合 |
| 后置条件 | 在一个制定价格回合中，制定价格 |
| Finance.export | 语法 | public void export(long id) |
| 前置条件 | 启动一个统计报表回合 |
| 后置条件 | 在一个统计报表回合中，导出excel表格 |
| Finance.getIncomeInfo | 语法 | public IncomeBillVO getIncomeInfo(long id) |
| 前置条件 | 启动一个结算管理回合 |
| 后置条件 | 在一个结算管理回合中，返回收款信息 |
| Finance.addIncomeInfo | 语法 | public void addIncomeInfo(IncomeBillVO income) |
| 前置条件 | 启动一个结算管理回合 |
| 后置条件 | 在一个结算管理回合中，将收款信息添加到收款单 |
| Finance.incomeBill | 语法 | public void incomeBill() |
| 前置条件 | 收款信息已全部输入 |
| 后置条件 | 生成收款单 |
| Finance.payBill | 语法 | public void payBill(PayBillVO pay) |
| 前置条件 | 启动一个成本管理回合 |
| 后置条件 | 生成新付款单 |
| Finance.costBenifit | 语法 | public CostProfitVO costProfit(String date) |
| 前置条件 | 启动一个成本管理回合 |
| 后置条件 | 返回成本收益信息（总收入、总支出、总利润） |
| Finance.costBenifitBill | 语法 | public void costBenifitBill() |
| 前置条件 | 获得成本收益信息 |
| 后置条件 | 生成成本收益表 |

1. 变更后

|  |  |  |
| --- | --- | --- |
| **提供的接口（供接口）** | | |
| Approval.approval | 语法 | public boolean approval(DocumentVO document) |
| 前置条件 | 启动一个审批单据回合 |
| 后置条件 | 在一个审批单据回合中，判定单据是否通过 |
| Price.getPrice | 语法 | public double getPrice(Express type) |
| 前置条件 | 启动一个制定价格回合 |
| 后置条件 | 在一个制定价格回合中，返回指定类型的快递价格 |
| Price.pricing | 语法 | public void pricing(double price) |
| 前置条件 | 启动一个制定价格回合 |
| 后置条件 | 在一个制定价格回合中，制定价格 |
| NumericalStatement  .export | 语法 | public void export(long id) |
| 前置条件 | 启动一个统计报表回合 |
| 后置条件 | 在一个统计报表回合中，导出excel表格 |
| Settlement.getIncomeInfo | 语法 | public IncomeBillVO getIncomeInfo(long id) |
| 前置条件 | 启动一个结算管理回合 |
| 后置条件 | 在一个结算管理回合中，返回收款信息 |
| Settlement  .addIncomeInfo | 语法 | public void addIncomeInfo(IncomeBillVO income) |
| 前置条件 | 启动一个结算管理回合 |
| 后置条件 | 在一个结算管理回合中，将收款信息添加到收款单 |
| Settlement.incomeBill | 语法 | public void incomeBill() |
| 前置条件 | 收款信息已全部输入 |
| 后置条件 | 生成收款单 |
| Cost.payBill | 语法 | public void payBill(PayBillVO pay) |
| 前置条件 | 启动一个成本管理回合 |
| 后置条件 | 生成新付款单 |
| Cost.costBenifit | 语法 | public CostProfitVO costProfit(String date) |
| 前置条件 | 启动一个成本管理回合 |
| 后置条件 | 返回成本收益信息（总收入、总支出、总利润） |
| Cost.costBenifitBill | 语法 | public void costBenifitBill() |
| 前置条件 | 获得成本收益信息 |
| 后置条件 | 生成成本收益表 |

1. OrderBl模块的接口规范
2. 变更原因

与FinanceBl发生变更的原因类似，OrderBl模块包含了众多任务，导致高度耦合，我们将OrderBl模块分成了

1. 变更前

|  |  |  |
| --- | --- | --- |
| **提供的接口（供接口）** | | |
| Order.addSender | 语法 | public ResultMessage addSender(SenderVO sender) |
| 前置条件 | 启动一个订单管理回合 |
| 后置条件 | 在一个订单管理回合中，增加寄件人的信息 |
| Order.addReceiver | 语法 | public ResultMessage addReceiver(ReceiverVO receiver) |
| 前置条件 | 启动一个订单管理回合 |
| 后置条件 | 在一个订单管理回合中，增加收件人的信息 |
| Order.chooseType | 语法 | public void chooseType(Express type) |
| 前置条件 | 已添加收件人、寄件人信息 |
| 后置条件 | 设定快递的类型 |
| Order.addGoods | 语法 | public ResultMessage addGoods(GoodsVO goods) |
| 前置条件 | 启动一个订单管理回合 |
| 后置条件 | 在一个订单管理回合中，增加货物信息 |
| Order.getMoney | 语法 | public double getMoney() |
| 前置条件 | 已添加收件人、寄件人和货物信息 |
| 后置条件 | 计算并返回货物运费 |
| Order.getTotal | 语法 | public double getTotal() |
| 前置条件 | 已计算所有单件货物的运费 |
| 后置条件 | 返回此订单管理回合中需要支付的总额 |
| Order.getTime | 语法 | public String getTime() |
| 前置条件 | 已添加收件人、寄件人和货物信息 |
| 后置条件 | 返回预估时间 |
| Order.createID | 语法 | public long createID() |
| 前置条件 | 已添加收件人、寄件人和货物信息 |
| 后置条件 | 生成条形码 |
| Order.endOrder | 语法 | public void endOrder() |
| 前置条件 | 已支付 |
| 后置条件 | 结束此次订单管理回合，持久化更新涉及的领域对象的数据 |
| Order.addOrder | 语法 | public void addOrder(long id) |
| 前置条件 | 启动一个接收/派件/记录收款单/车辆装车管理/中转接收/装运管理回合 |
| 后置条件 | 在订单列表中添加订单 |
| Order.createArrivalNote | 语法 | public void createArrivalNote(ArrivalVO arrivalInfo) |
| 前置条件 | 在一个接收回合中，完成所有信息的输入 |
| 后置条件 | 结束此次接收回合，持久化更新涉及的领域对象和数据 |
| Order.createDeliveryNote | 语法 | public void createDeliveryNote(DeliveryVO deliveryInfo) |
| 前置条件 | 在一个派件回合中，完成所有信息的输入 |
| 后置条件 | 结束此次派件回合，持久化更新涉及的领域对象和数据 |
| Order.createIncomeBill | 语法 | public void createIncomeBill(IncomeBillVO income) |
| 前置条件 | 在一个记录收款单回合中，完成所有信息的输入 |
| 后置条件 | 结束此次记录收款单回合，持久化更新涉及的领域对象和数据 |
| Order.createVehicleLoadNote | 语法 | public void createVehicleLoadNote(VehicleLoadVO vLoadvo) |
| 前置条件 | 在一个车辆装车管理回合中，完成所有信息的输入 |
| 后置条件 | 结束此次车辆装车管理回合，持久化更新涉及的领域对象和数据 |
| Order.createTransferNote | 语法 | public void createTransferNote(TransferVO transferInfo) |
| 前置条件 | 在一个中转接收回合中，完成所有信息的输入 |
| 后置条件 | 结束中转接收回合，持久化更新涉及的领域对象和数据 |
| Order.createLoad Note | 语法 | public void createLoadNote(LoadVO loadInfo) |
| 前置条件 | 在一个装运管理回合中，完成所有信息的输入 |
| 后置条件 | 结束装运管理回合，持久化更新涉及的领域对象和数据 |

1. 变更后

|  |  |  |
| --- | --- | --- |
| **提供的接口（供接口）** | | |
| Order.addSender | 语法 | public ResultMessage addSender(SenderVO sender) |
| 前置条件 | 启动一个订单管理回合 |
| 后置条件 | 在一个订单管理回合中，增加寄件人的信息 |
| Order.addReceiver | 语法 | public ResultMessage addReceiver(ReceiverVO receiver) |
| 前置条件 | 启动一个订单管理回合 |
| 后置条件 | 在一个订单管理回合中，增加收件人的信息 |
| Order.chooseType | 语法 | public void chooseType(Express type) |
| 前置条件 | 已添加收件人、寄件人信息 |
| 后置条件 | 设定快递的类型 |
| Order.addGoods | 语法 | public ResultMessage addGoods(GoodsVO goods) |
| 前置条件 | 启动一个订单管理回合 |
| 后置条件 | 在一个订单管理回合中，增加货物信息 |
| Order.getMoney | 语法 | public double getMoney() |
| 前置条件 | 已添加收件人、寄件人和货物信息 |
| 后置条件 | 计算并返回货物运费 |
| Order.getTotal | 语法 | public double getTotal() |
| 前置条件 | 已计算所有单件货物的运费 |
| 后置条件 | 返回此订单管理回合中需要支付的总额 |
| Order.getTime | 语法 | public String getTime() |
| 前置条件 | 已添加收件人、寄件人和货物信息 |
| 后置条件 | 返回预估时间 |
| Order.createID | 语法 | public long createID() |
| 前置条件 | 已添加收件人、寄件人和货物信息 |
| 后置条件 | 生成条形码 |
| Order.endOrder | 语法 | public void endOrder() |
| 前置条件 | 已支付 |
| 后置条件 | 结束此次订单管理回合，持久化更新涉及的领域对象的数据 |
| AddOrder.addOrder | 语法 | public void addOrder(long id) |
| 前置条件 | 启动一个接收/派件/记录收款单/车辆装车管理/中转接收/装运管理回合 |
| 后置条件 | 在订单列表中添加订单 |
| Receipt.createArrivalNote | 语法 | public void createArrivalNote(ArrivalVO arrivalInfo) |
| 前置条件 | 在一个接收回合中，完成所有信息的输入 |
| 后置条件 | 结束此次接收回合，持久化更新涉及的领域对象和数据 |
| Sending.createDeliveryNote | 语法 | public void createDeliveryNote(DeliveryVO deliveryInfo) |
| 前置条件 | 在一个派件回合中，完成所有信息的输入 |
| 后置条件 | 结束此次派件回合，持久化更新涉及的领域对象和数据 |
| ReceiptRecord.createIncomeBill | 语法 | public void createIncomeBill(IncomeBillVO income) |
| 前置条件 | 在一个记录收款单回合中，完成所有信息的输入 |
| 后置条件 | 结束此次记录收款单回合，持久化更新涉及的领域对象和数据 |
| VehicleLoad.createVehicleLoadNote | 语法 | public void createVehicleLoadNote(VehicleLoadVO vLoadvo) |
| 前置条件 | 在一个车辆装车管理回合中，完成所有信息的输入 |
| 后置条件 | 结束此次车辆装车管理回合，持久化更新涉及的领域对象和数据 |
| Transfer.createTransferNote | 语法 | public void createTransferNote(TransferVO transferInfo) |
| 前置条件 | 在一个中转接收回合中，完成所有信息的输入 |
| 后置条件 | 结束中转接收回合，持久化更新涉及的领域对象和数据 |
| Load.createLoad Note | 语法 | public void createLoadNote(LoadVO loadInfo) |
| 前置条件 | 在一个装运管理回合中，完成所有信息的输入 |
| 后置条件 | 结束装运管理回合，持久化更新涉及的领域对象和数据 |

1. 添加选择包装类型接口
2. 变更原因

遗漏了选择不同包装类型的需求

1. 变更后

添加接口

|  |  |  |
| --- | --- | --- |
| Order.choosePackageType | 语法 | public void choosePackageType(Packing type) |
| 前置条件 | 已添加收件人、寄件人信息 |
| 后置条件 | 设定包装的类型 |

1. 修改生成单据接口
2. 变更原因

生成单据的接口无需参数

原定设计为传入单据信息（如到达单需传入ArrivalVO），根据单据信息直接生成单据，但这样会导致数据传输频繁。变更后，接口实现类直接持有单据信息的引用，生成单据时，无需再传入参数。

1. 变更前

因涉及变更较多且内容类似，故以createArrivalNote为例描述变更前后的情形

|  |  |  |
| --- | --- | --- |
| Receipt.createArrivalNote | 语法 | public void createArrivalNote(ArrivalVO arrivalInfo) |
| 前置条件 | 在一个接收回合中，完成所有信息的输入 |
| 后置条件 | 结束此次接收回合，持久化更新涉及的领域对象和数据 |

1. 变更后

|  |  |  |
| --- | --- | --- |
| Receipt.createArrivalNote | 语法 | public void createArrivalNote() |
| 前置条件 | 在一个接收回合中，完成所有信息的输入 |
| 后置条件 | 结束此次接收回合，持久化更新涉及的领域对象和数据 |

1. 删除Data层的finish接口
2. 变更原因

变更前每次使用数据库时，需要先打开数据库连接，使用后关闭数据库连接，因为这一操作大多相同，于是将这两个操作单独封装成Connect接口，并提供打开数据库连接的接口

1. 变更前

以OrderData为例：

|  |  |  |
| --- | --- | --- |
| OrderDataService.finish | 语法 | public void finish() throws RemoteException |
| 前置条件 | 无 |
| 后置条件 | 结束持久化数据库的使用 |

1. 变更后

删除该类接口

1. 添加Connect接口
2. 变更原因

如上所述，打开与关闭数据库操作重复内容较多，故单独封装为接口

1. 变更后

添加接口

|  |  |  |
| --- | --- | --- |
| **提供的服务（供接口）** | | |
| Connect. getPreparedStatement | 语法 | public PreparedStatement getPreparedStatement(String sql) |
| 前置条件 | sql语句符合语句规范 |
| 后置条件 | 返回相应的PreparedStatement对象 |
| Connect.getResultSet | 语法 | public ResultSet getResultSet(String sql) |
| 前置条件 | sql为查询数据库操作 |
| 后置条件 | 返回表示查询结果ResultSet对象 |
| Connect.closeConnection | 语法 | public void closeConnection() |
| 前置条件 | 已打开数据库连接 |
| 后置条件 | 关闭数据库连接 |