

面向对象分析与设计

Object Oriented Analysis and Design

——商铺模块设计

Design of Shop Module

邱明 博士

厦门大学信息学院

mingqiu@xmu.edu.cn

2023年秋季学期

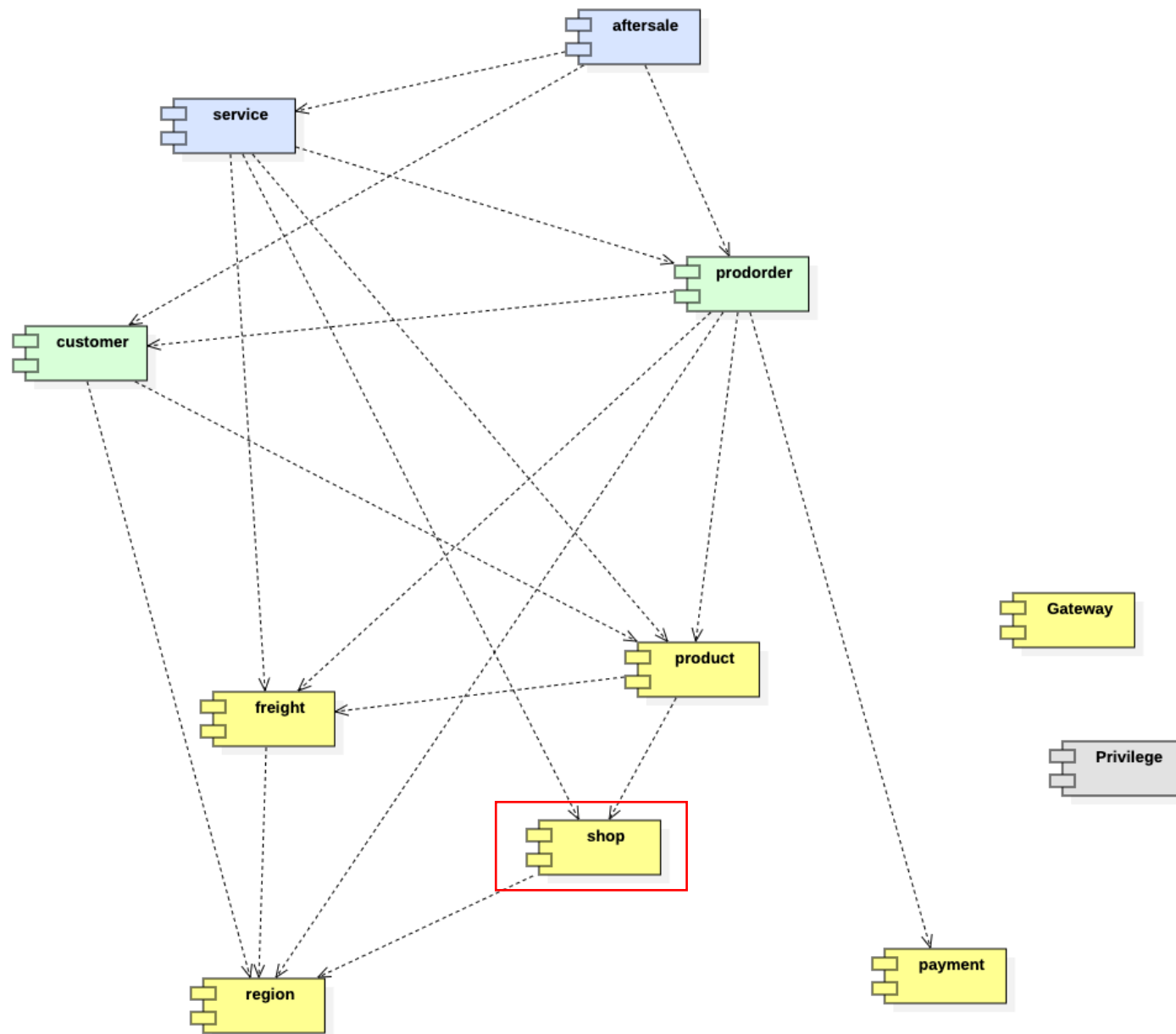
1. 静态模型设计

Static Model Design



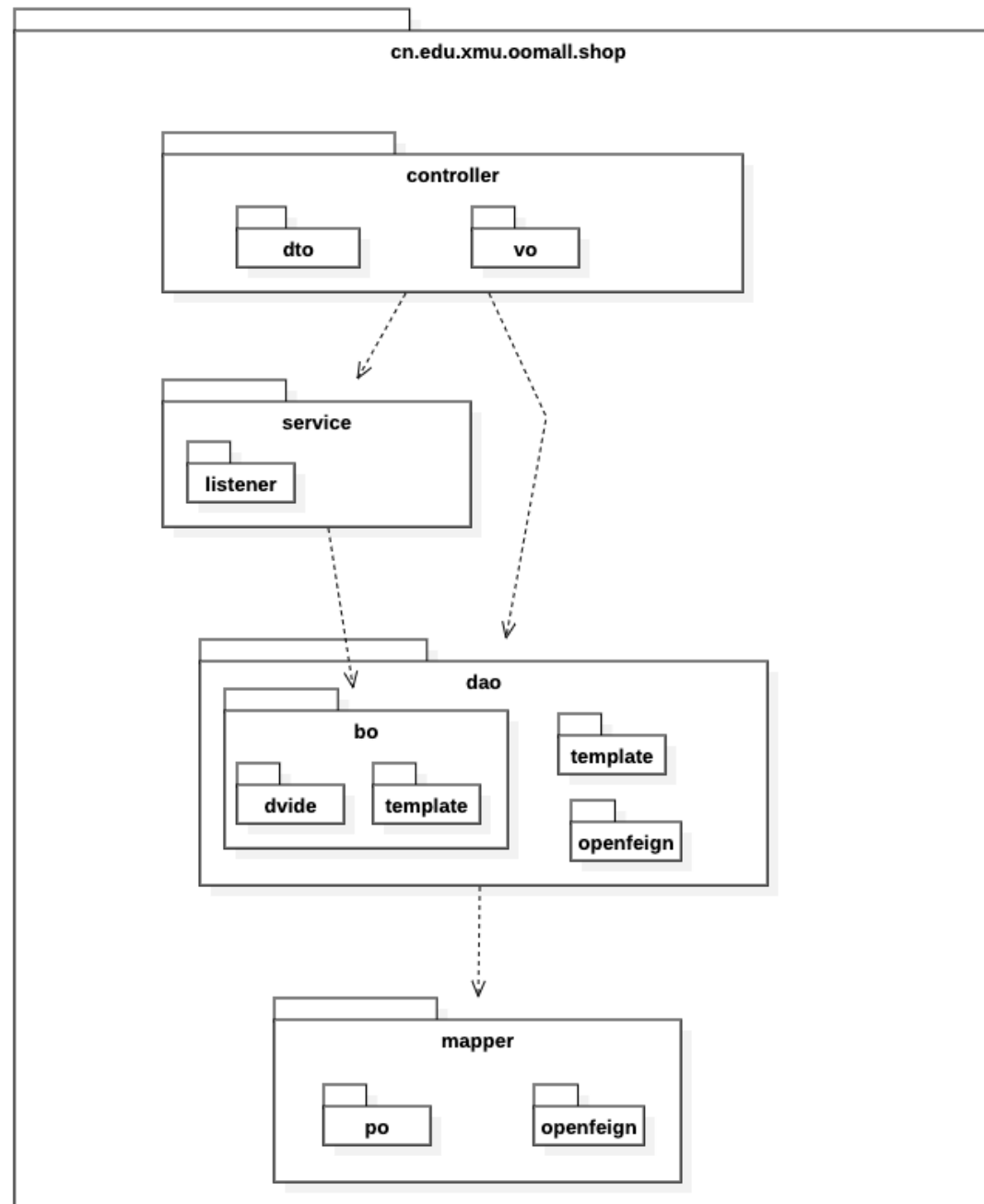
1.1 总体结构

Architectural Structure



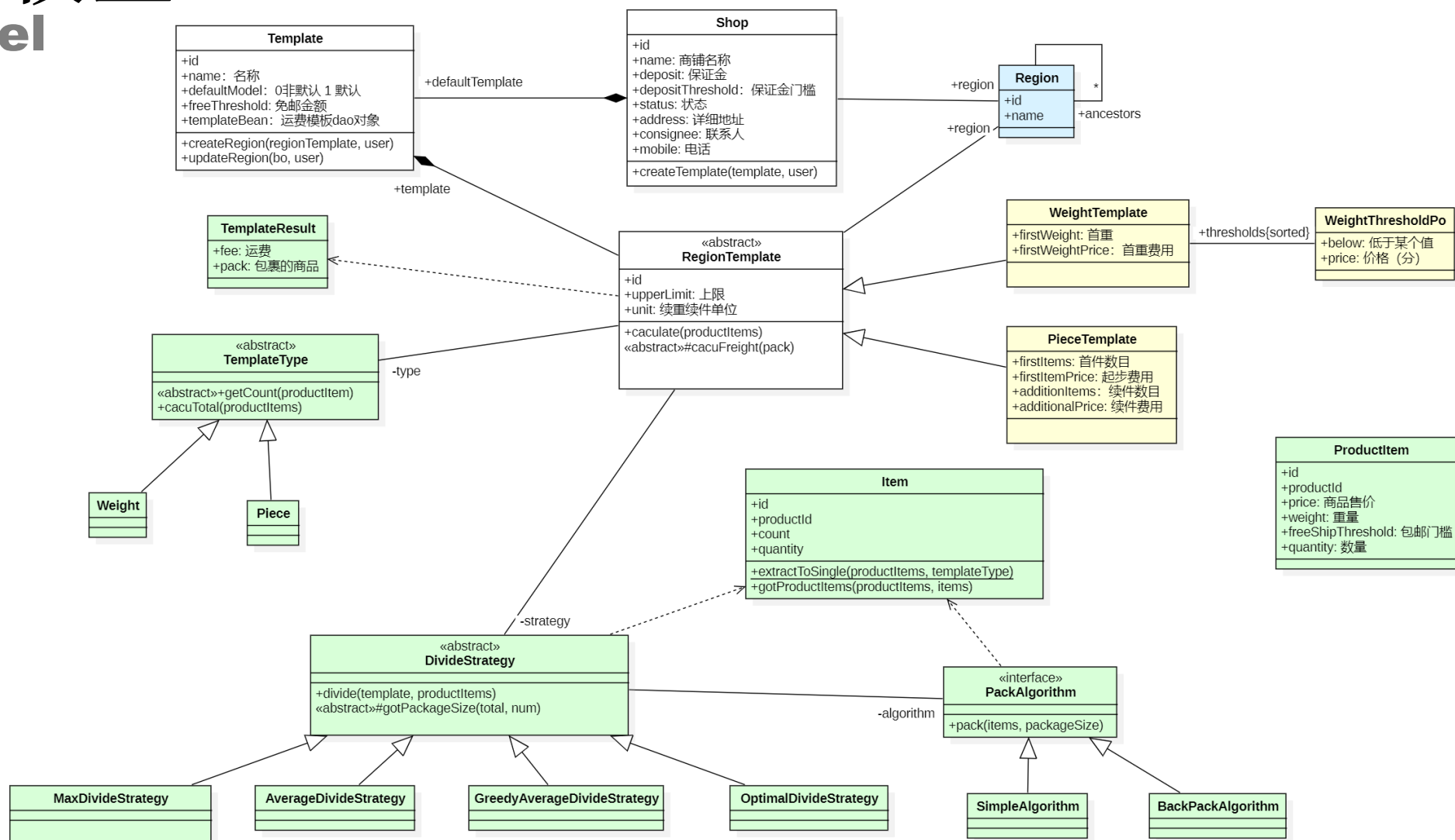
1.2 包结构

Design Goals



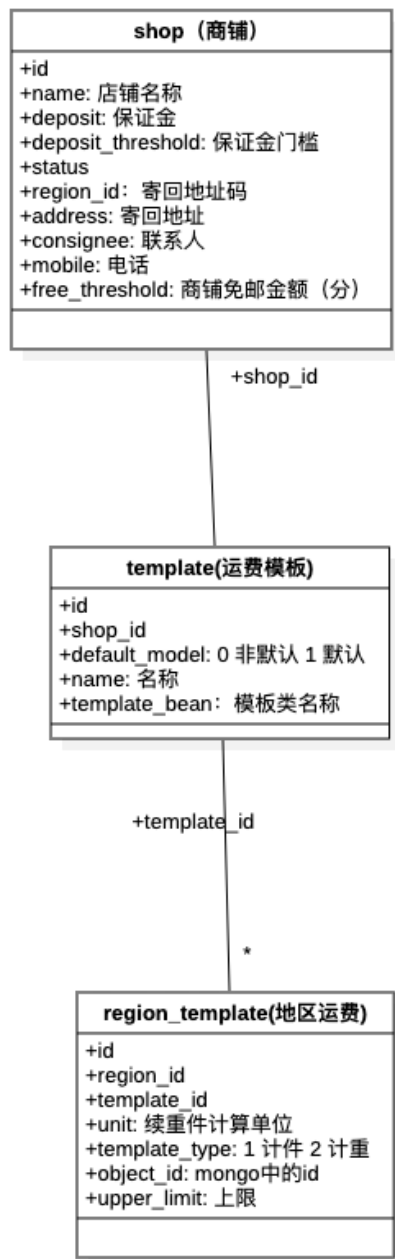
1.3 对象模型

Object Model



1.4 数据库设计

Database Design



2. 动态模型设计

Dynamic Model Design



2.1 定义模板明细

<https://app.swaggerhub.com/apis/mingqcn/OOMALL/1.3.2#/shop/postWeightTemplates>

POST

/shops/{shopId}/templates/{id}/regions/{rid}/weighttemplate 管理员定义重量模板明细。

✎ ↶ ↷

- 需要登录
- templateType:2 tempalteDao:weightTemplateDao.

Try it out

Parameters

Name	Description
authorization * required string (header)	用户token <input type="text" value="authorization"/>
shopId * required integer (\$int32) (path)	商铺id <input type="text" value="shopId"/>
id * required integer (\$int64) (path)	运费模板id <input type="text" value="id"/>
rid * required integer (\$int64) (path)	地区id <input type="text" value="rid"/>
freightModelInfo * required object (body)	运费模板资料。 Example Value Model <pre>{ "unit": 0, "upperLimit": 0, "firstWeight": 0, "firstWeightFreight": 0, "thresholds": [{ "below": 0, "price": 0 }]}</pre> Parameter content type <div>application/json</div>

Responses

Response content type application/json

Code	Description
195	该运费模板中该地区(id=%d)已经定义
default	成功 Example Value Model <pre>{ "errno": 0, "errmsg": "成功"}</pre>



2.1 定义模板明细

<https://app.swaggerhub.com/apis/mingqcn/OOMALL/1.3.2#/shop/postPieceTemplates>



厦门大学信息学院

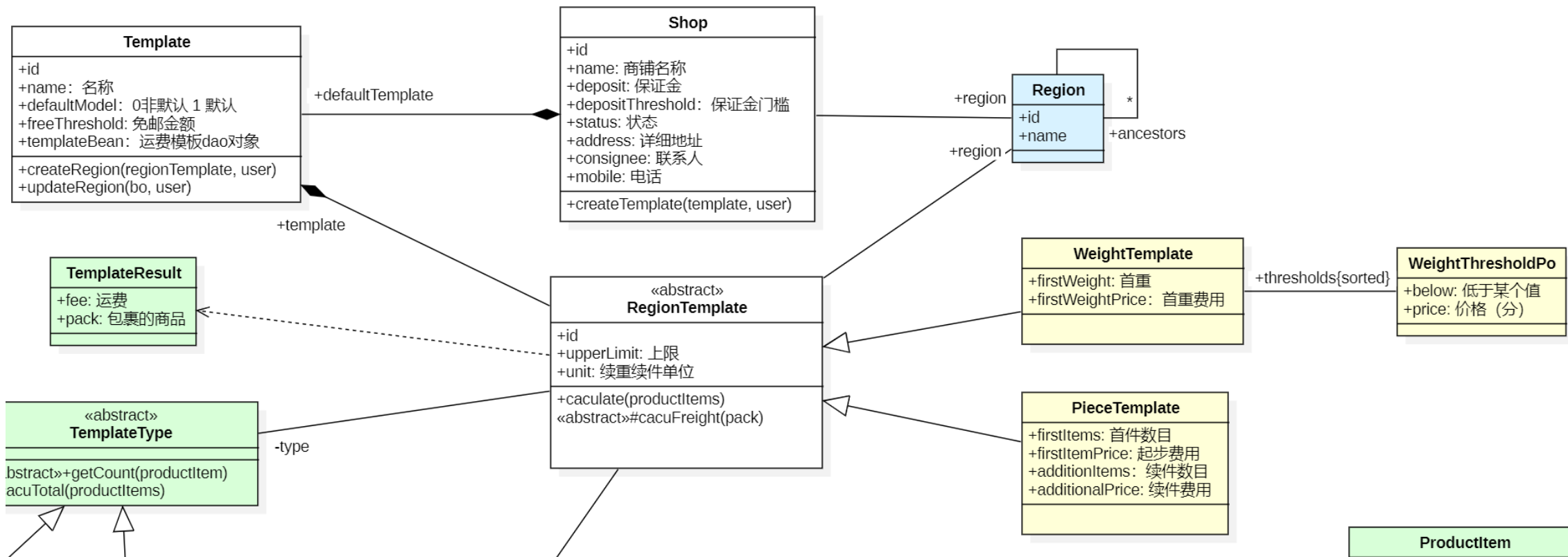
<div><div>POST</div><div>/shops/{shopId}/templates/{id}/regions/{rid}/piecetemplates</div><div>管理员定义件数模板明细。</div></div>								
<div><div><div><div>需要登录</div><div>templateType:1 templateDao:pieceTemplateDao.</div></div></div></div>								
<div><div>Parameters</div><div><div><div><div><div><div>Try it out</div></div></div></div><div><table><thead><tr><th>Name</th><th>Description</th></tr></thead><tbody><tr><td><div><div>authorization ★ required</div><div>string</div><div>(header)</div></div><div>用户token</div><div><div>authorization</div></div></td><td rowspan="5"><div><div>shopId ★ required</div><div>integer (\$int32)</div><div>(path)</div></div><div>商铺id</div><div><div>shopId</div></div></td></tr><tr><td><div><div>id ★ required</div><div>integer (\$int64)</div><div>(path)</div></div><div>模板id</div><div><div>id</div></div></td></tr><tr><td><div><div>rid ★ required</div><div>integer (\$int64)</div><div>(path)</div></div><div>地区id</div><div><div>rid</div></div></td></tr><tr><td><div><div>freightModelInfo ★ required</div><div>object</div><div>(body)</div></div><div>运费模板资料。</div><div><div><div>Example Value Model</div><div><pre>{ "unit": 0, "upperLimit": 0, "firstItem": 0, "firstItemPrice": 0, "additionalItems": 0, "additionalItemsPrice": 0 }</pre></div></div></div><div><div>Parameter content type</div><div>application/json</div></div></td></tr></tbody></table></div></div></div></div>		Name	Description	<div><div>authorization ★ required</div><div>string</div><div>(header)</div></div> <div>用户token</div> <div><div>authorization</div></div>	<div><div>shopId ★ required</div><div>integer (\$int32)</div><div>(path)</div></div> <div>商铺id</div> <div><div>shopId</div></div>	<div><div>id ★ required</div><div>integer (\$int64)</div><div>(path)</div></div> <div>模板id</div> <div><div>id</div></div>	<div><div>rid ★ required</div><div>integer (\$int64)</div><div>(path)</div></div> <div>地区id</div> <div><div>rid</div></div>	<div><div>freightModelInfo ★ required</div><div>object</div><div>(body)</div></div> <div>运费模板资料。</div> <div><div><div>Example Value Model</div><div><pre>{ "unit": 0, "upperLimit": 0, "firstItem": 0, "firstItemPrice": 0, "additionalItems": 0, "additionalItemsPrice": 0 }</pre></div></div></div> <div><div>Parameter content type</div><div>application/json</div></div>
Name	Description							
<div><div>authorization ★ required</div><div>string</div><div>(header)</div></div> <div>用户token</div> <div><div>authorization</div></div>	<div><div>shopId ★ required</div><div>integer (\$int32)</div><div>(path)</div></div> <div>商铺id</div> <div><div>shopId</div></div>							
<div><div>id ★ required</div><div>integer (\$int64)</div><div>(path)</div></div> <div>模板id</div> <div><div>id</div></div>								
<div><div>rid ★ required</div><div>integer (\$int64)</div><div>(path)</div></div> <div>地区id</div> <div><div>rid</div></div>								
<div><div>freightModelInfo ★ required</div><div>object</div><div>(body)</div></div> <div>运费模板资料。</div> <div><div><div>Example Value Model</div><div><pre>{ "unit": 0, "upperLimit": 0, "firstItem": 0, "firstItemPrice": 0, "additionalItems": 0, "additionalItemsPrice": 0 }</pre></div></div></div> <div><div>Parameter content type</div><div>application/json</div></div>								
<div><div>Responses</div><div><div>Response content type</div><div>application/json</div></div></div>								
<div><div><div>Code</div><div>195</div><div>default</div></div></div>	<div><div><div>Description</div><div>该运费模板中该地区(id=%d)已经定义</div><div>成功</div></div><div><div><div>Example Value Model</div><div><pre>{ "errno": 0, "errmsg": "成功" }</pre></div></div></div></div>							

2.1 定义模板明细

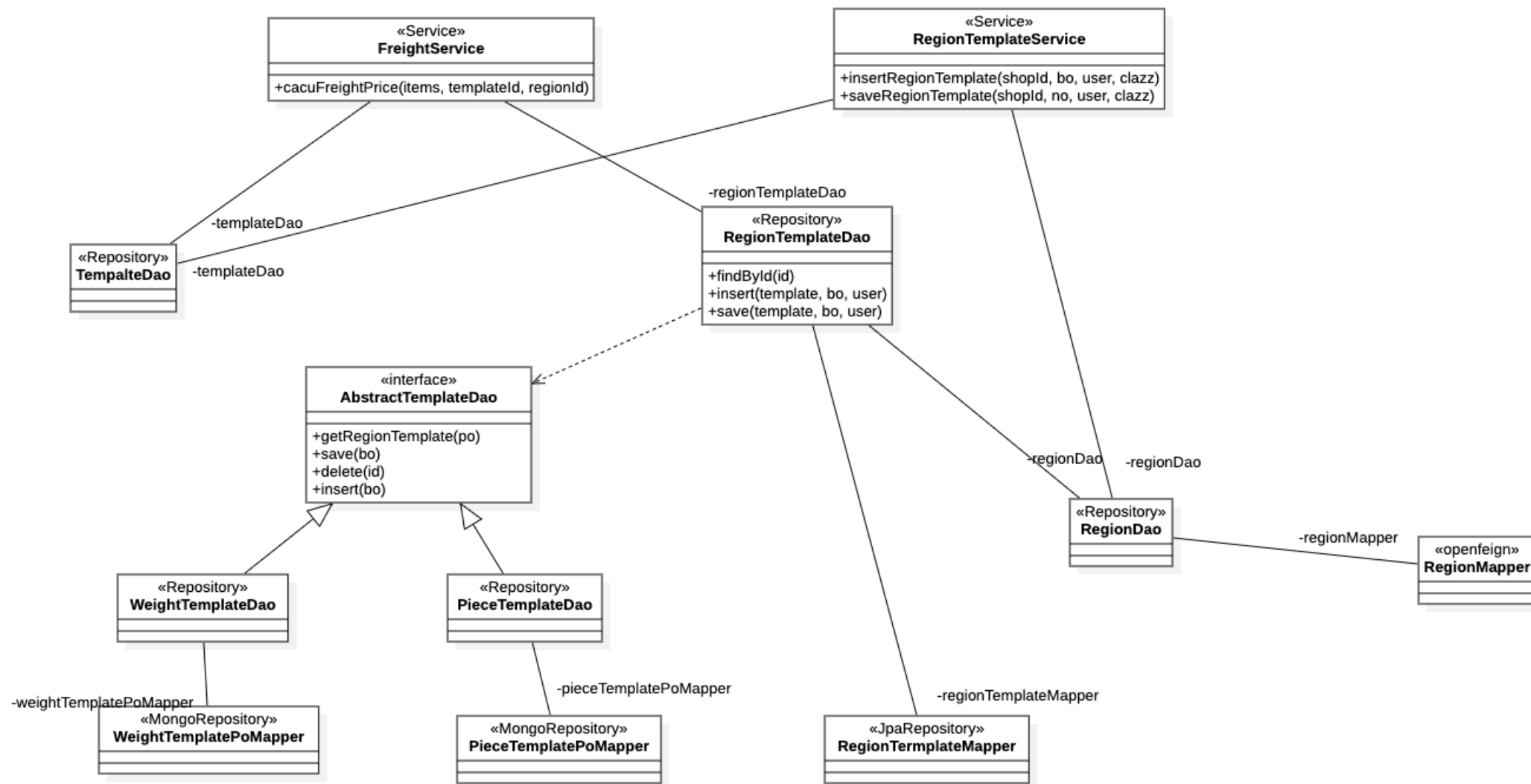


1.3 对象模型

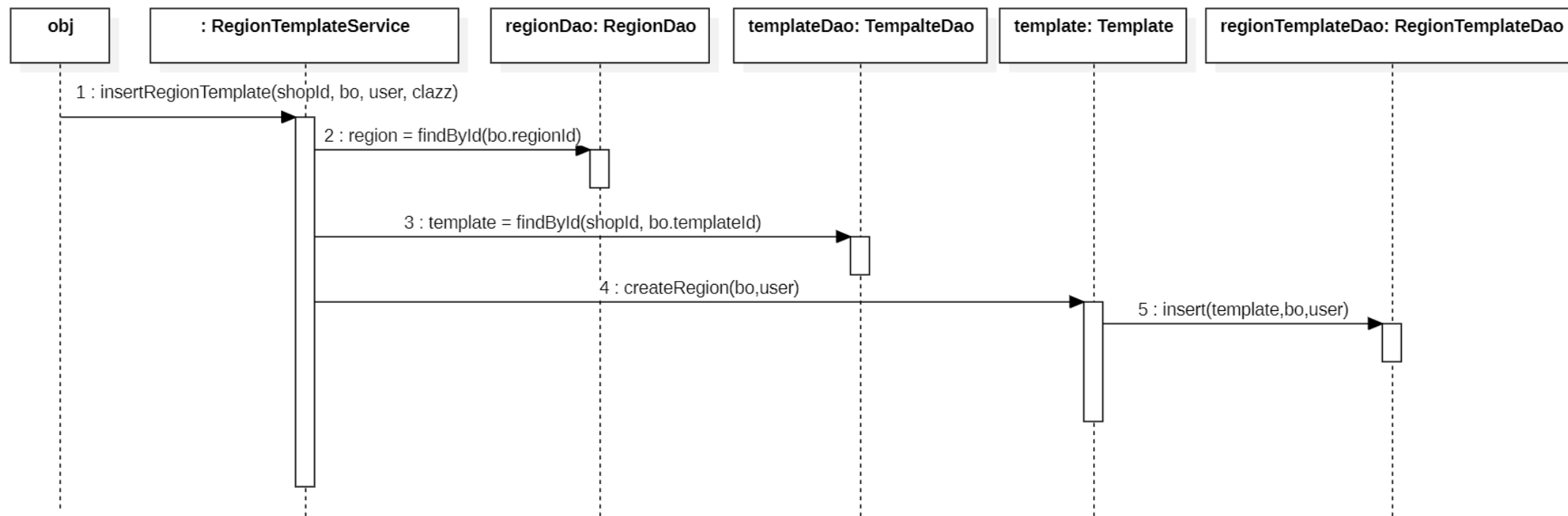
Object Model



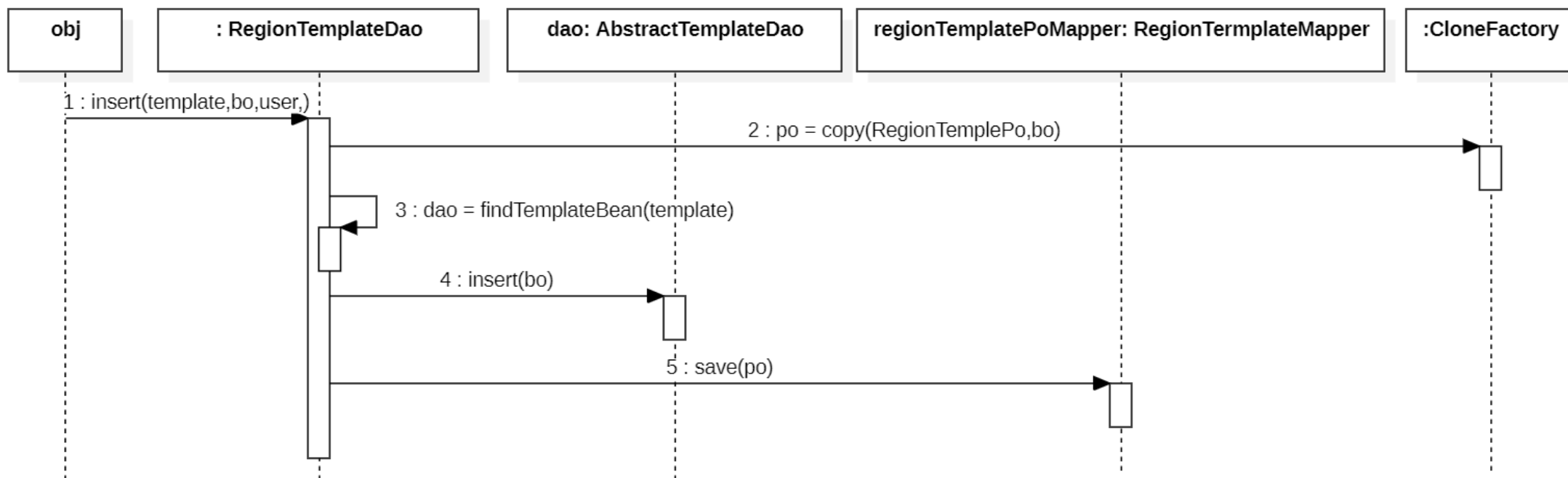
2.1 定义模板明细



2.1 定义模板明细



2.1 定义模板明细



2.2 修改模板明细

<https://app.swaggerhub.com/apis/mingqcn/OOMALL/1.3.2#/shop/putWeightTemplates>



PUT /shops/{shopId}/templates/{id}/regions/{rid}/weighttemplate 管理员修改重量模板明细。

- 需要登录
- 管理员调用本 API 定义默认运费模板。

Try it out

Name	Description
authorization * required string (header)	用户token <input type="text" value="authorization"/>
shopId * required integer(\$int32) (path)	商铺id <input type="text" value="shopId"/>
id * required integer(\$int64) (path)	运费模板id <input type="text" value="id"/>
rid * required integer(\$int64) (path)	地区id <input type="text" value="rid"/>
freightModelInfo * required object (body)	运费模板资料。 Example Value Model <pre>{ "unit": 0, "upperLimit": 0, "firstWeight": 0, "firstWeightFreight": 0, "thresholds": [{ "below": 0, "price": 0 }]}</pre> <div>Parameter content type <div>application/json</div></div>

Responses

Response content type application/json

Code	Description
default	成功 Example Value Model <pre>{ "errno": 0, "errmsg": "成功"}</pre>

2.2 修改模板明细

<https://app.swaggerhub.com/apis/mingqcn/OOMALL/1.3.2#/shop/putPieceTemplates>



PUT

/shops/{shopId}/templates/{id}/regions/{rid}/piecetemplates 管理员修改件数模板。

📄 ↶ ↷

• 需要登录

Parameters

Try it out

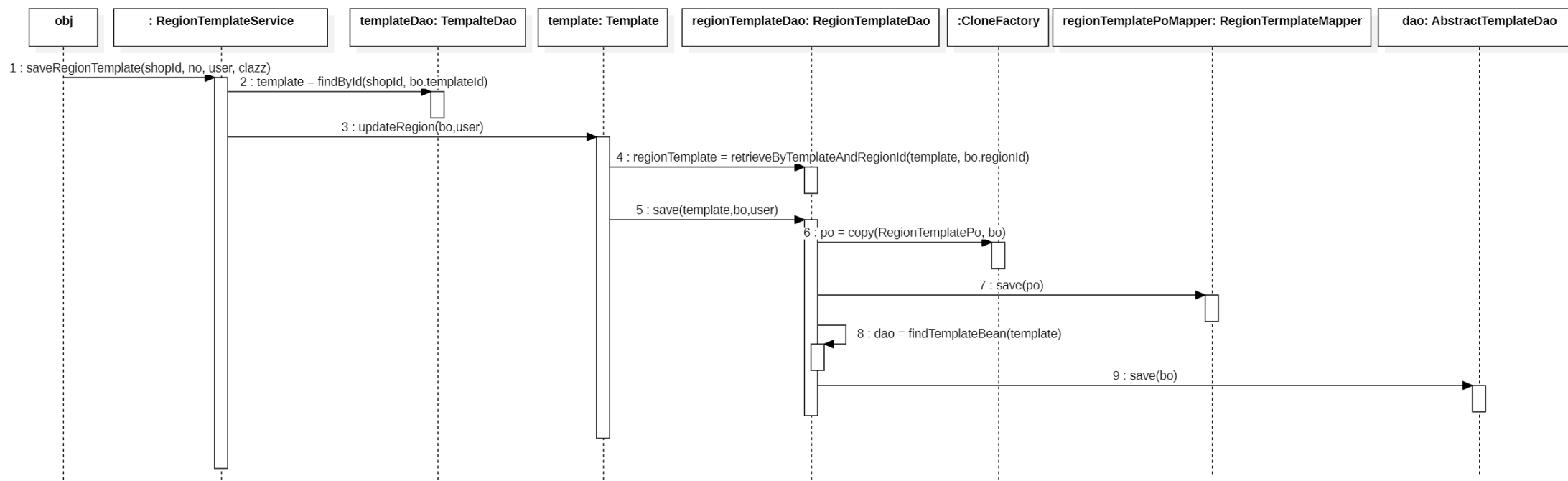
Name	Description
authorization * required string (header)	用户token <input type="text" value="authorization"/>
shopId * required integer (\$int32) (path)	商铺id <input type="text" value="shopId"/>
id * required integer (\$int64) (path)	模板id <input type="text" value="id"/>
rid * required integer (\$int64) (path)	地区id <input type="text" value="rid"/>
freightModelInfo * required object (body)	运费模板资料。 <div>Example Value Model</div> <pre>{ "unit": 0, "upperLimit": 0, "firstItem": 0, "firstItemPrice": 0, "additionalItems": 0, "additionalItemsPrice": 0}</pre> <div>Parameter content type <input type="text" value="application/json"/></div>

Responses

Response content type

Code	Description
default	成功 <div>Example Value Model</div> <pre>{ "errno": 0, "errmsg": "成功"}</pre>

2.2 修改模板明细



2.3 计算一批商品的运费

<https://app.swaggerhub.com/apis/mingqcn/OOMALL/1.3.2#/shop/postFreightPrice>

- 当超过包裹上限时，不同的打包策略会有不同的运费
 - 上限打包
 - 均匀打包，
 - 费用最优打包



POST /internal/templates/{id}/regions/{rid}/freightprice 内部API-计算一批商品的运费。

Try it out

Name	Description
authorization * required string (header)	用户token <input type="text" value="authorization"/>
id * required integer(\$int64) (path)	模板id <input type="text" value="id"/>
rid * required integer(\$int64) (path)	地区id <input type="text" value="rid"/>
items * required array[object] (body)	订单商品的订货详情 Example Value Model <pre>[{ "orderId": 0, "productId": 0, "quantity": 0, "weight": 0 }]</pre> <div>Parameter content type application/json</div>

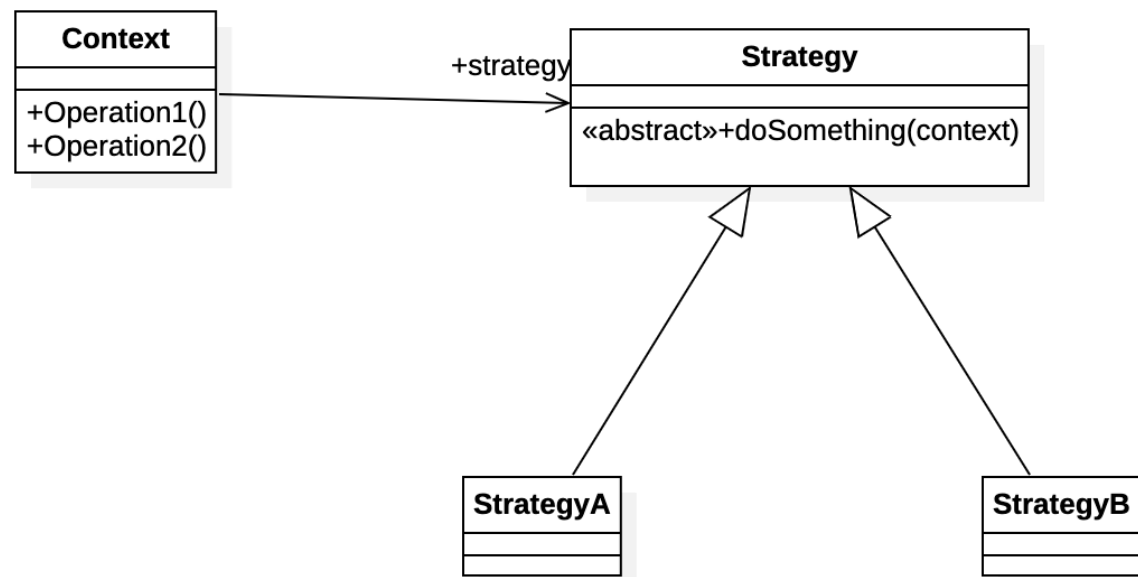
Responses

Response content type application/json

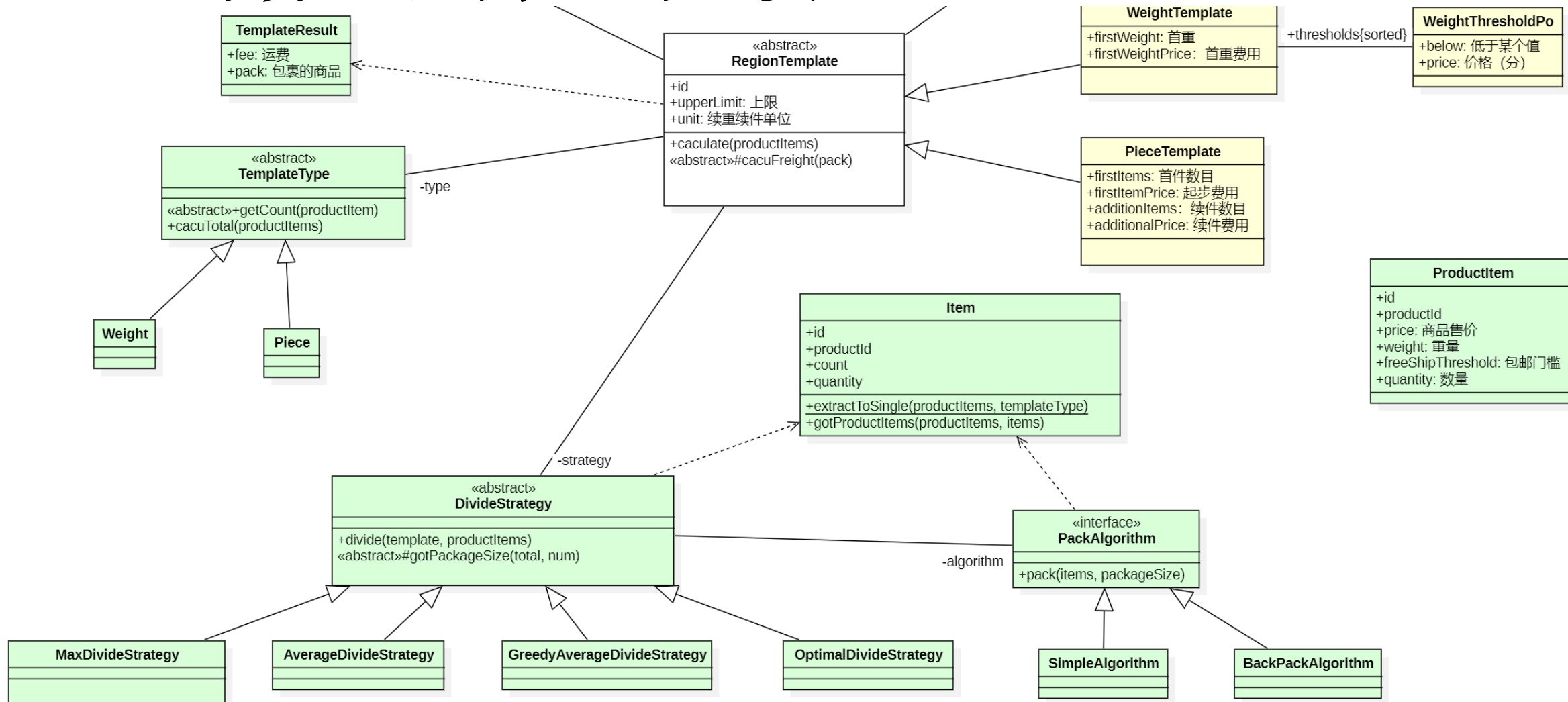
Code	Description
default	成功 Example Value Model <pre>{ "errno": 0, "errmsg": "成功", "data": { "freightPrice": 0, "pack": [{ "pack": [{ "orderId": 0, "productId": 0, "quantity": 0 }], "price": 0 }] } }</pre>

2.3 计算一批商品的运费

- 策略模式 (Strategy)
 - 定义一系列算法，把它们封装在对象内部，并且可以任意替换。

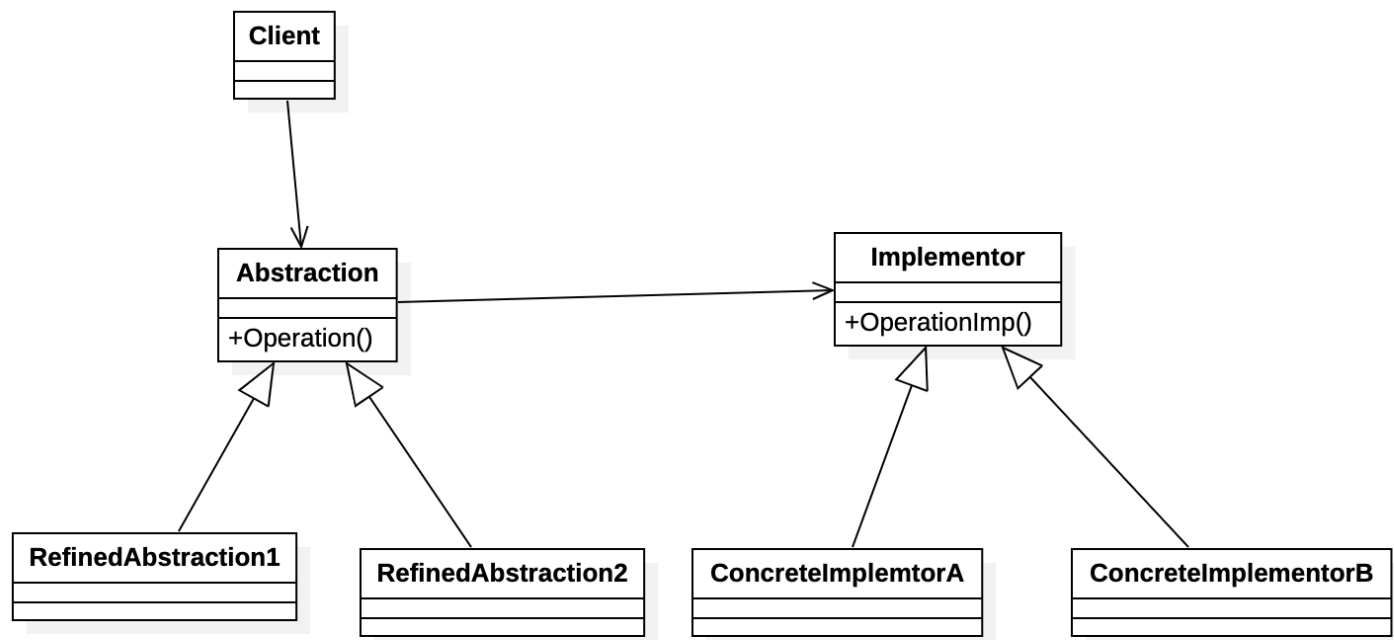


2.3 计算一批商品的运费

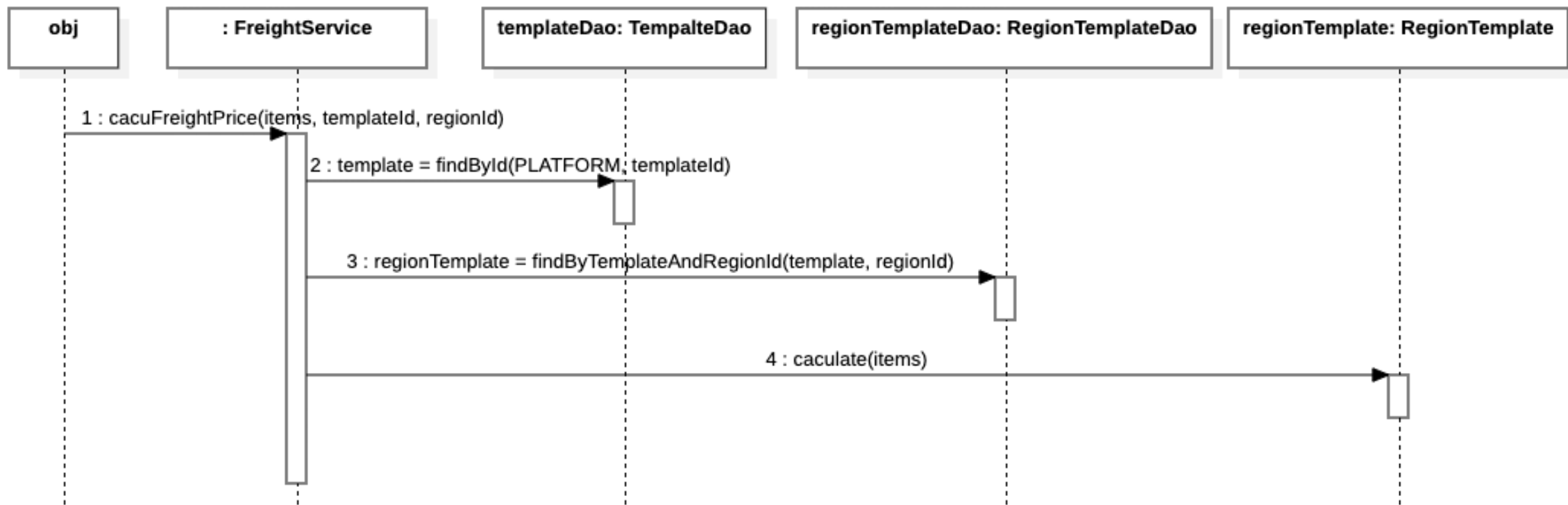


2.3 计算一批商品的运费

- 桥接器模式（Bridge）
 - 将问题分解成独立的两个部分，任意组合形成最后的解决方案



2.3 计算一批商品的运费



2.3 计算一批商品的运费

