3 Sell

This chapter presents three compound components:
- Product-sale management, for businesses that collect payment by issuing invoices to customers and (eventually) getting paid.
- Cash-sale management, for businesses that collect payment at the time of a sale
- Customer-account management

2.1 Sell / Product-Sale Management

What. For flexibility, we treat a product as a material resource with some added responsibilities. In this way, we can take any material resource and turn it into a product (this happens in some industries).

A product can be a product in the traditional sense, a service, or a combination of both.

Product-sale management supports the selling of products (goods, services, or combination of both)—on an invoicing basis, rather than on a "cash and carry" basis (as in cash-sales management).

Scope. Product-sale management starts with sales and ends with invoicing.

Steps. Define product types and products. Make a sale to a customer. Ship products. Invoice the customer. Record the delivery of products; track and resolve delivery-problem reports. Make agreements and assessments.

Links. Deduct quantity from inventory (link with material-resource management; it interacts with inventory management). Post invoice totals (accounting management).

Mirror images. In product-sales management, we move things out of the business on an invoicing basis (from us to a customer). In material-resource management, we move things into the business on an invoicing basis (from a supplier to us).

Components. The components within product-sale management are (Figure 3-1):
- Product
- Sale to customer
- Shipment to customer
- Delivery to customer
- Invoice to customer
- Product agreement
- Product assessment

Moment-intervals. The main moment-intervals for product-sale management are (Figure 3-2):
- Product price
- Sale to customer
- Shipment to customer

- Delivery to customer
- Delivery-problem report
- Invoice to customer
- Discount agreement
- Commission agreement
- Cost and overhead allocation
- Marketing study
- Sales forecast
- Geographic-region assignment

Interactions. The components work together to get things done. An example of inter-component interaction, "calculate direct commissions for a sales rep", is shown in Figure 3-2. A sender asks yellow sales rep to calculate its commissions, the ones coming from his own sales (called "direct" commissions). A sales-rep object asks each of its pink sales to build a list of product-sale details. Next, a sales-rep object asks each of its pink commission objects to calculate direct commissions. A commission object then matches its product descriptions and quantities with the sales details, looking for a valid match, then computes the commission for the sales of that product. Some commission objects might not be linked to a product description, in which case that commission applies across all of the product-sale details. At the end, the sales rep returns its results to the sender.

Expansion. One could expand this compound component by adding components to support pre-sales activities, including customer and prospect rankings, test marketing activities and results, and feature tracking for future products. One could also expand it with post-sale activities including service.

| ProductSaleMgmt.SummaryInPink | ProductSaleMgmt.Product | ProductSaleMgmt.ShipmentToCustomer | ProductSaleMgmt.ProductAgreement |

| ProductSaleMgmt.SaleToCustomer | ProductSaleMgmt.DeliveryToCustomer | ProductSaleMgmt.ProductAssessment |

| ProductSaleMgmt.InvoiceToCustomer |
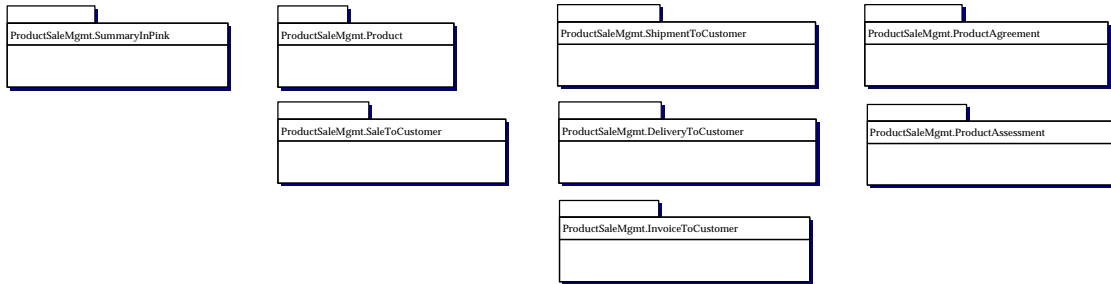
Figure 3-1. Product-sale management components.

<<moment-interval>>
...Product.ProductPrice

<<moment-interval>>
...SaleToCustomer.SaleToCustomer

0..*
back order, auto-repeat

1..*
0..1

0..*

<<moment-interval>>
...ShipmentToCustomer.ShipmentToCustomer

1
0..*

1..*

<<moment-interval>>
...DeliveryToCustomer.DeliveryToCustomer

0..*

<<moment-interval>>
...DeliveryToCustomer.DeliveryProblemReport

0..*

<<moment-interval>>
...InvoiceToCustomer.InvoiceToCustomer

0..1
adjustment

0..1

<<moment-interval>>
...ProductAgreement.DiscountAgreement

<<moment-interval>>
...ProductAgreement.CommissionAgreement

<<moment-interval>>
...ProductAssessment.CostAndOverheadAllocation

<<moment-interval>>
...ProductAssessment.MarketingStudy

<<moment-interval>>
...ProductAssessment.SalesForecast

<<moment-interval>>
...ProductAssessment.GeogRegionAssignment

Figure 3-2. Summary in pink.

aSender	aSalesRep	aSale	aCommissionAgreement	aDetail
	RelMgmt.PersonRole.SalesRep	...SaleToCustomer.SaleToCustomer	...CommissionAgreement	...CommissionAgreementDetail

1: calcDirectCommissions

FOR each
sale

2: listProductSaleDetails

3: calcDirectCommissionsForProductSaleDetails

FOR each
product-
sale detail

4: calcDirectCommissionForProductSaleDetail
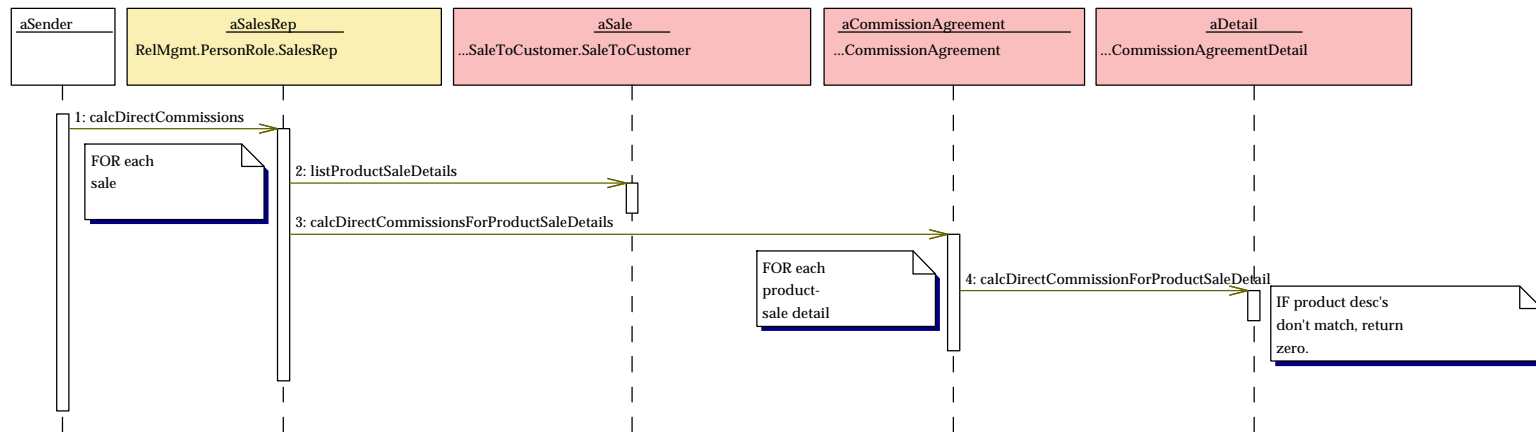
IF product desc's
don't match, return
zero.

Figure 3-3. Calculate direct commissions for a sales rep.

2.1.1 Sell / Product-Sale Management / Product

Guided tour. The product component has two central classes: a green product and a blue product description.

Product. A green product is something that a business sells, is individually identifiable (it has a serial number), and is something that must be individually tracked. If a product were not individually identifiable, you would not need a green thing; instead, you could use a quantity of a blue catalog-entry-like description. Moreover, if something were individually identifiable yet you needed to track specific quantities, then again a quantity of a blue catalog-entry-like description would be sufficient. If you need to track more than just quantity; *e.g.,* specific serialized items, then you need to manage a collection of these item. The green product could have a collection of serial numbers, or a blue catalog-entry-like description could have a collection of green serialized products.

A green product has required links to a green material resource and a blue product description. A blue product description has a required link to a blue material-resource description.

For example, consider a specific Ford F-100 truck, identifiable by its serial number, called a vehicle identification number. A green product linked to a green material-resource represents it.

Now consider a standard catalog-entry that applies to any Ford F-100 on a dealer's lot: the manufacturer is Ford, the model name is F-100, the vehicle type is truck, and so on. A blue product linked to a blue product description represents it.

A green product links to some yellow product being sold roles.

Product description. This is a catalog-entry-like description of a kind of material resource. A blue material-resource description is the main description; it links to a number of other supporting blue context-specific supplemental descriptions that one can add as needed. Notice that the blue tax category links to its applicable green geographic regions.

Other components use certain quantities of a product description. For example, if someone requests 20 Ford F-100 trucks, then a component might include a request detail with a quantity of 20, linked to a blue product description that applies to each Ford F-100.

Product price. A pink product price sets a price for a quantity and price unit of measure, applicable for an interval of time. It links to blue product description(s) or to green products. It also links to the yellow pricer responsible for setting that price.

You have some modeling choices when it comes to price. One, you can model it as an attribute in the green product (for example, the price for that red Ferrari, that one right there!) or a blue description (for example, the price of a Snicker's bar of a particular size). However, if you want to track that price in the past (for trend analysis), in the present (to make sales), and in the future (planning for forthcoming price changes) then yes you need a pink moment-interval for that.

 Tip. An attribute? Or something more? If you need an attribute value, use an attribute. If you need to track the change in that value over time (past, present, future), use a pink moment-interval. If you need to set that value once and then apply it to other objects as a standard, use a blue description.

Product catalog. A green catalog is a collection of blue product descriptions. If the catalog were a catalog of one-of-a-kind collectibles, it would link to green product(s) too. And if a catalog were to have catalog-specific prices, then a catalog would link to its pink prices, with those prices linking to its product descriptions.

Tip. Track values for each link? Add a pink moment-interval. Just need to categorize links? Use this simpler approach: label the endpoint of the link with the categories.

Methods. Key methods include calculate price for quantity and unit of measure, list catalog entries with catalog-specific prices, and verify availability of a quantity.

The product component is shown in Figure 3-4.

Interactions. The "calculate price for quantity and unit of measure" sequence is shown in Figure 3-5. A sender asks a blue product description to calculate its price, given a quantity and a unit of measure. The product description asks each of its pink product price(s) to calculate its price for a quantity and unit of measure—and goes with whatever is the best price. (In a retail system, the "best price" might be the lowest price. In an insurance system, it might be the highest price. Insurance app developers know what we mean! A business always looks different when standing on the other side of the counter.) A product price checks that it's valid for the date, gets the price, gets the quantity, gets the price unit of measure, does the math, and returns its result. At the end, the product description returns the price to the sender.

Another interesting interaction sequence is "list catalog entries with catalog-specific prices", shown in Figure 3-6. A sender asks a green product catalog to list its catalog entries with catalog-specific prices. The product catalog asks each of its pink product prices for its corresponding blue product descriptions. Then it interacts with both the product price and the product descriptions, adding to its list of catalog entries with prices. At the end, the product catalog returns the list to the sender.
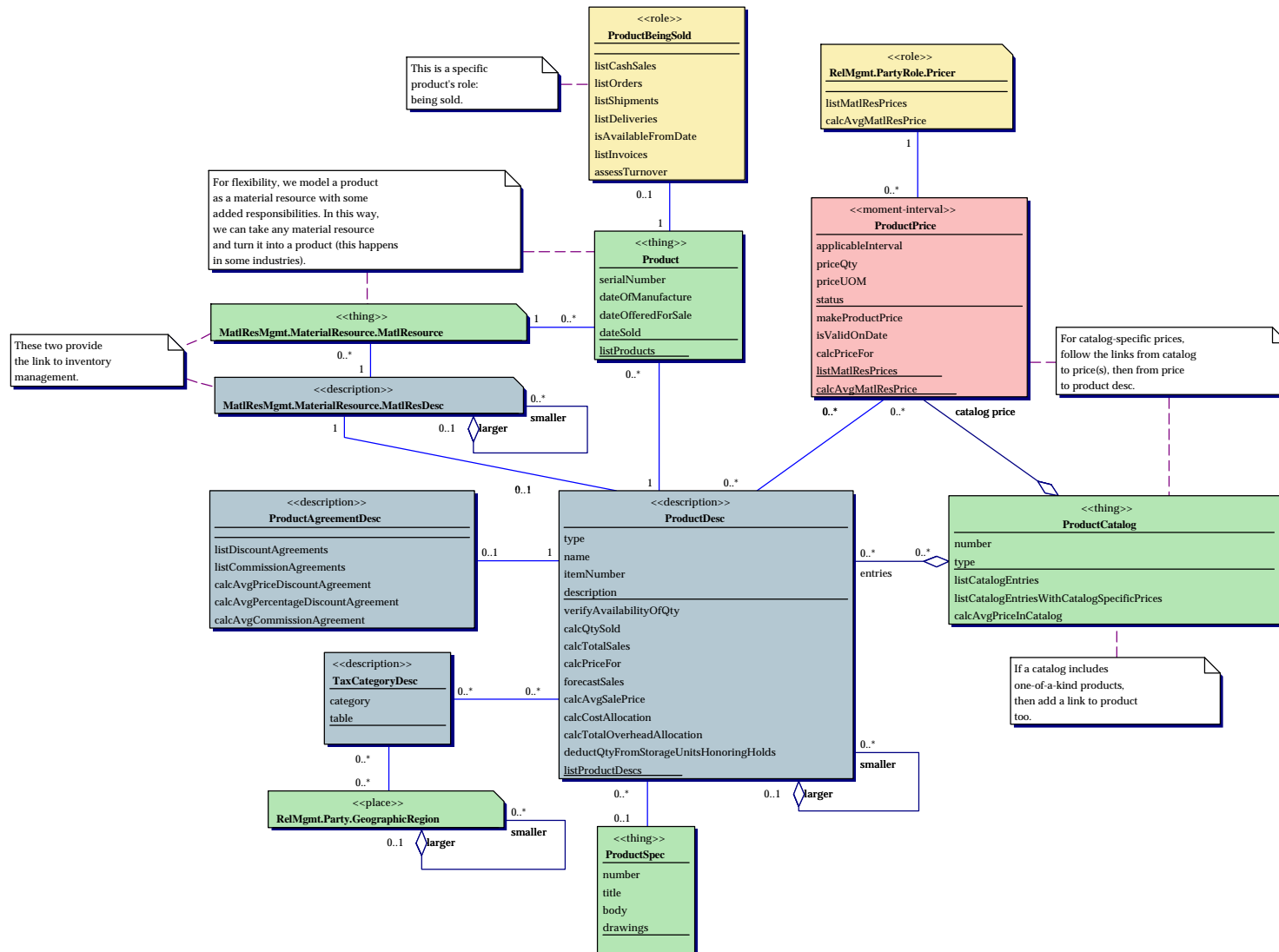
<<role>>
ProductBeingSold

listCashSales
listOrders
listShipments
listDeliveries
isAvailableFromDate
listInvoices
assessTurnover

This is a specific
product's role:
being sold.

<<role>>
RelMgmt.PartyRole.Pricer

listMatlResPrices
calcAvgMatlResPrice

1

0..*

0..1

1

For flexibility, we model a product
as a material resource with some
added responsibilities. In this way,
we can take any material resource
and turn it into a product (this happens
in some industries).

<<thing>>
Product

serialNumber
dateOfManufacture
dateOfferedForSale
dateSold
listProducts

<<moment-interval>>
ProductPrice

applicableInterval
priceQty
priceUOM
status
makeProductPrice
isValidOnDate
calcPriceFor
listMatlResPrices
calcAvgMatlResPrice

<<thing>>
MatlResMgmt.MaterialResource.MatlResource

1   0..*

0..*
1

These two provide
the link to inventory
management.

<<description>>
MatlResMgmt.MaterialResource.MatlResDesc

0..*

For catalog-specific prices,
follow the links from catalog
to price(s), then from price
to product desc.

0..*

0..*   0..*   **catalog price**

0..*

smaller

1   0..1   **larger**

0..1

1   0..*

<<description>>
ProductAgreementDesc

listDiscountAgreements
listCommissionAgreements
calcAvgPriceDiscountAgreement
calcAvgPercentageDiscountAgreement
calcAvgCommissionAgreement

0..1   1

<<description>>
ProductDesc

type
name
itemNumber
description
verifyAvailabilityOfQty
calcQtySold
calcTotalSales
calcPriceFor
forecastSales
calcAvgSalePrice
calcCostAllocation
calcTotalOverheadAllocation
deductQtyFromStorageUnitsHonoringHolds
listProductDescs

0..*   0..*
entries

<<thing>>
ProductCatalog

number
type
listCatalogEntries
listCatalogEntriesWithCatalogSpecificPrices
calcAvgPriceInCatalog

If a catalog includes
one-of-a-kind products,
then add a link to product
too.

0..*

smaller

0..1   **larger**

<<description>>
TaxCategoryDesc

category
table

0..*   0..*

0..*
0..*

<<place>>
RelMgmt.Party.GeographicRegion

0..*   smaller
0..1   **larger**

0..*   0..1

<<thing>>
ProductSpec

number
title
body
drawings

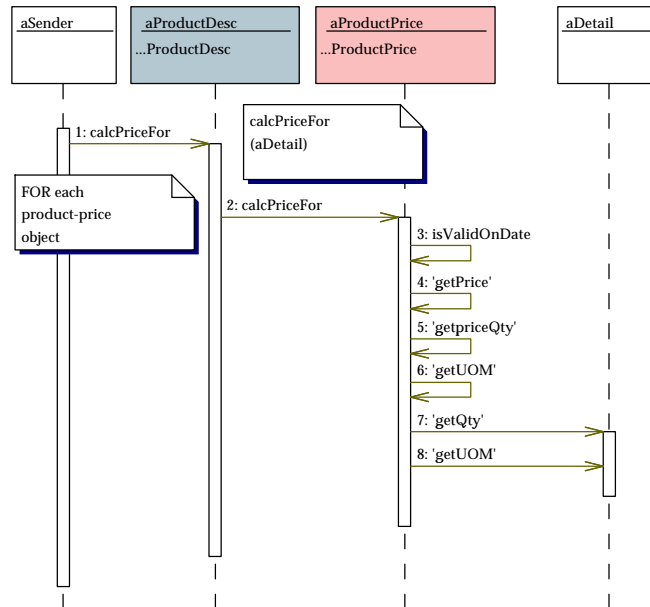*Figure 3-5. Product component.*

Figure 3-5. Calculate price for quantity and unit of measure.
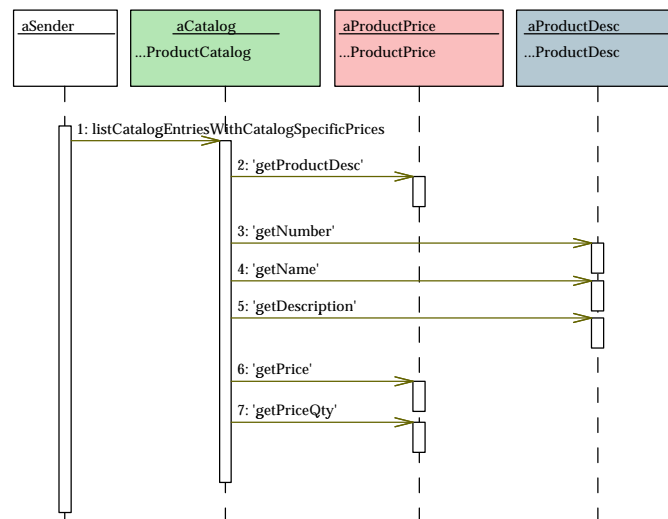

Figure 3-6. List catalog entries with catalog-specific prices.

2.1.2 Sell / Product-Sale Management / Sale to Customer

Guided tour. The sale-to-customer component has one pink moment-interval, sale to customer.

Sale to customer. A pink sale-to-customer links to two yellow roles: sales rep and customer. It links to a green (ship-to) address and a green terms-and-conditions object. It also links to pink sale-to-customer detail(s).

Sale-to-customer detail. A pink sale-to-customer detail specifies quantity, negotiated price, and status. It links to a blue product description. Or it might link to yellow product being sold roles, which in turn link to green products (in the product component). It might link to a green (ship-to) address.

Before and after. For sale to customer, the subsequent pink moment-interval is shipment to customer.

Methods. Key methods include make sale to customer, calculate the total of a sale, calculate quantity of a product description sold over an interval, and compare sale with deliveries.

The sale-to-customer component is shown in Figure 3-7.

Interactions. The "calculate quantity sold" sequence is shown in Figure 3-8. A sender asks a blue product description to calculate the quantity sold, passing along an applicable interval as an argument. The product description asks each of its pink sale details for its quantity within that interval. Each sale detail then asks its pink sale for its date, checks to make sure it's within the interval, then returns its amount (if within the interval) or zero (otherwise) to the product description. Ultimately, the product description returns its result to the sender.
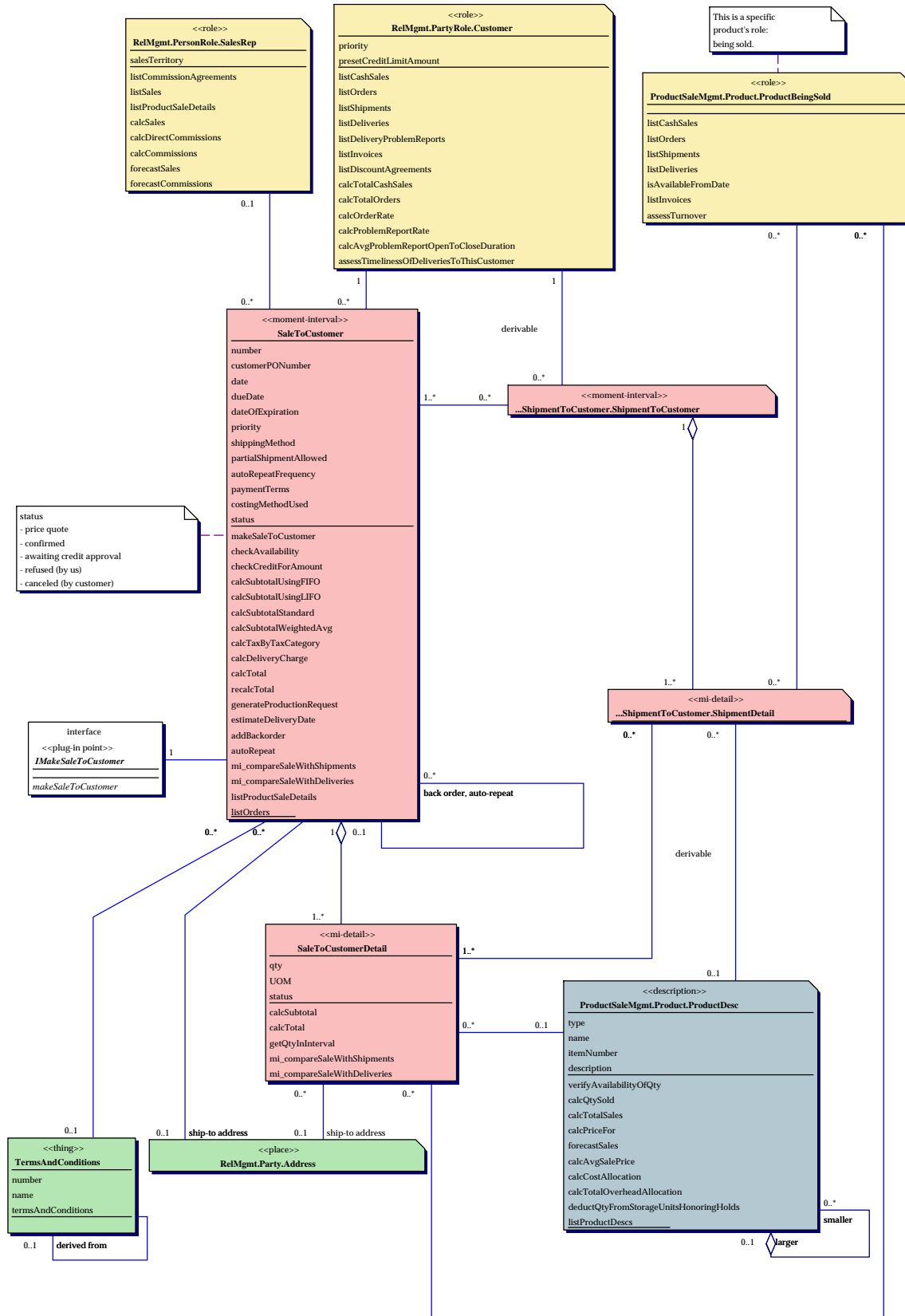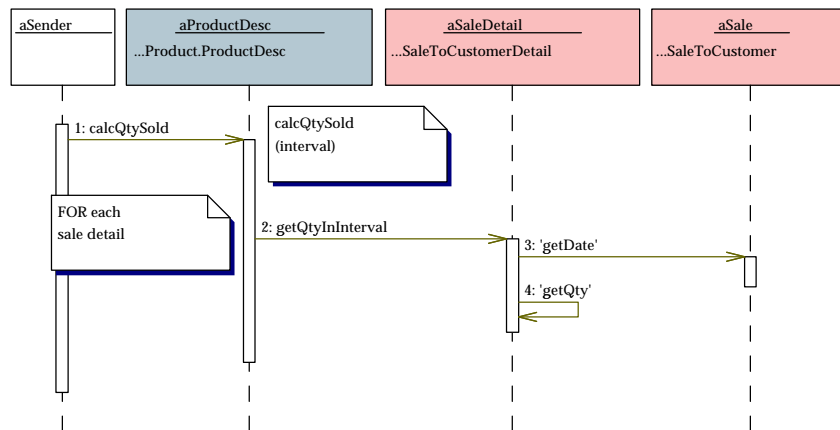
Figure 3-7. Sale-to-customer component.

Figure 3-8. Calculate quantity sold.

For more, read *Java Modeling in Color with UML.*
Order from amazon.com (search: java color).

2.2 Sell / Cash-Sale Management

For more, read *Java Modeling in Color with UML.*
Order from amazon.com (search: java color).

2.3 Sell / Customer-Account Management

For more, read *Java Modeling in Color with UML.*
Order from amazon.com (search: java color).