

A Business Rules Example

Building smart and flexible Customer Services

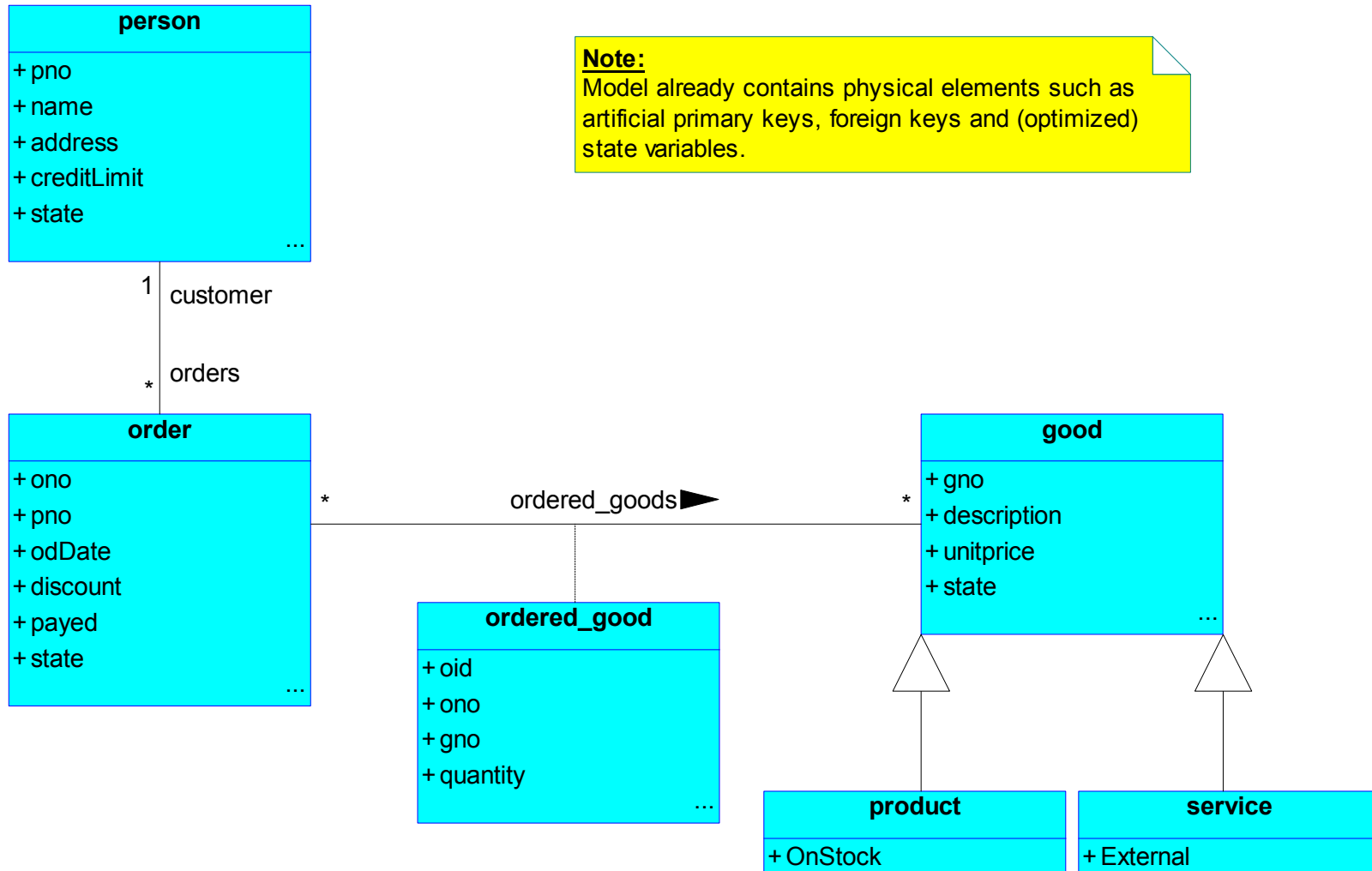
Markus Schacher

KnowGravity Inc, Badenerstrasse 808, 8048 Zürich

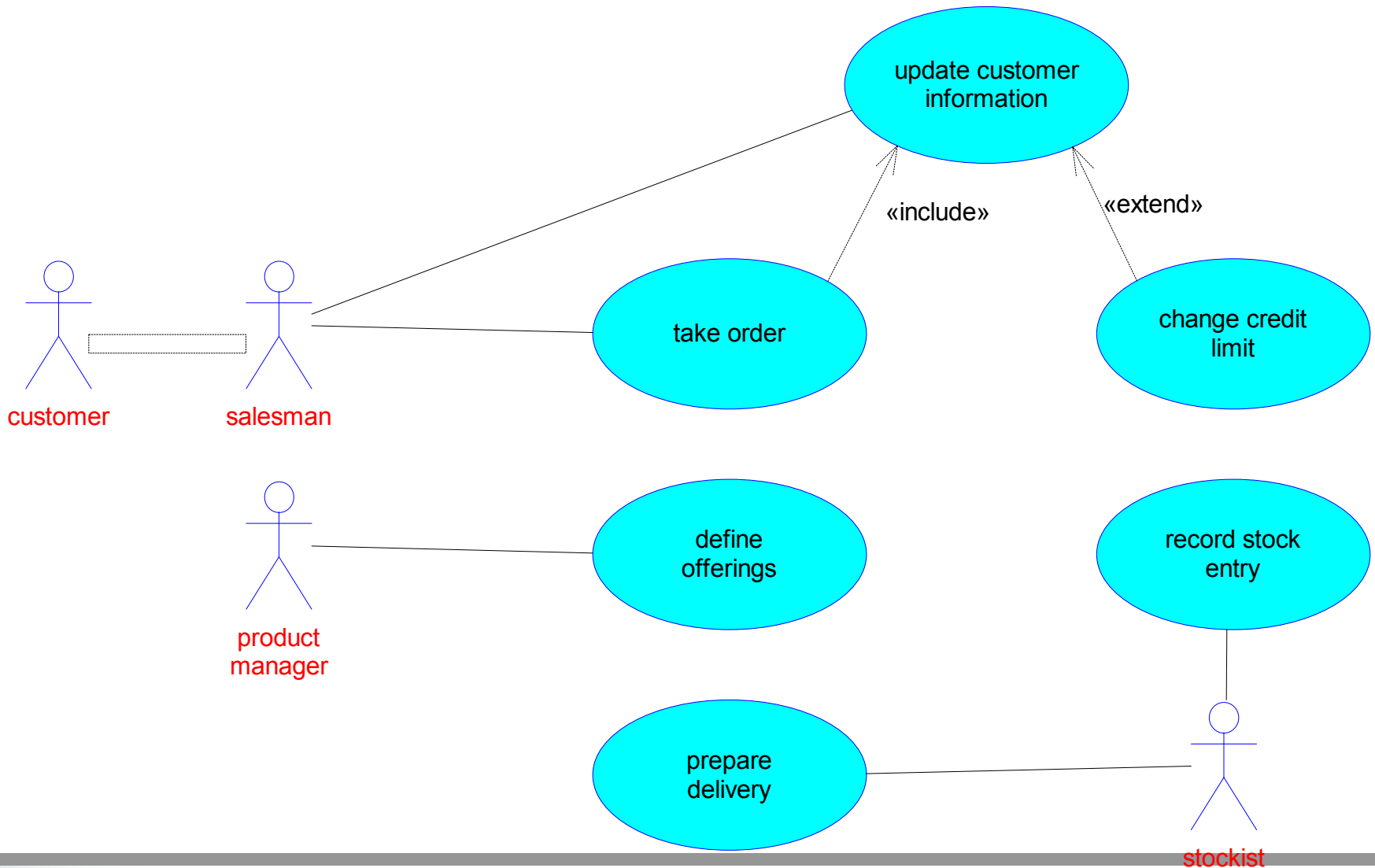
Tel. 01/434'20'00, Fax. 01/434'20'09

<http://www.knowgravity.com>

Business Objects



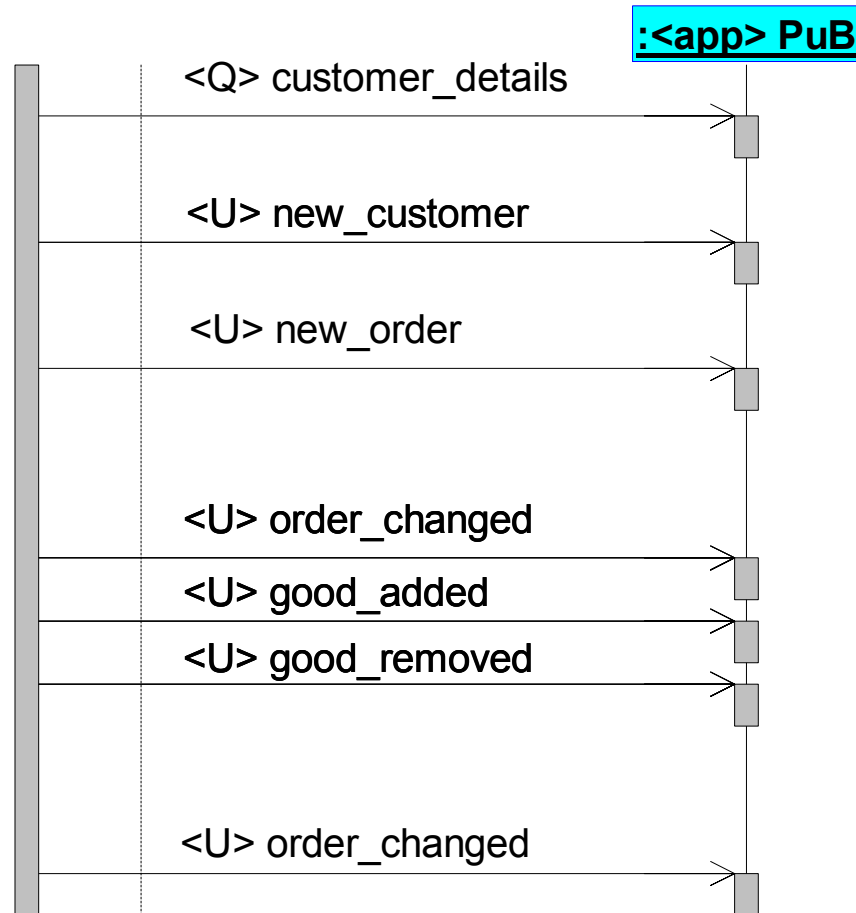
Use Case Model



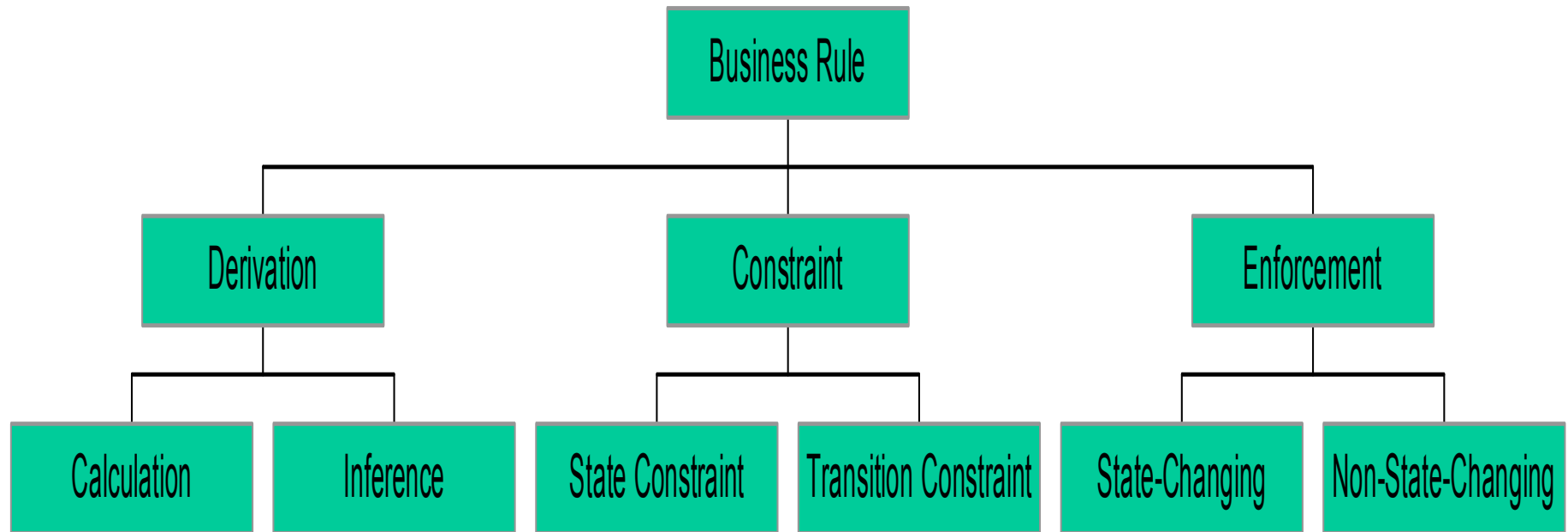
Use Case "take order"



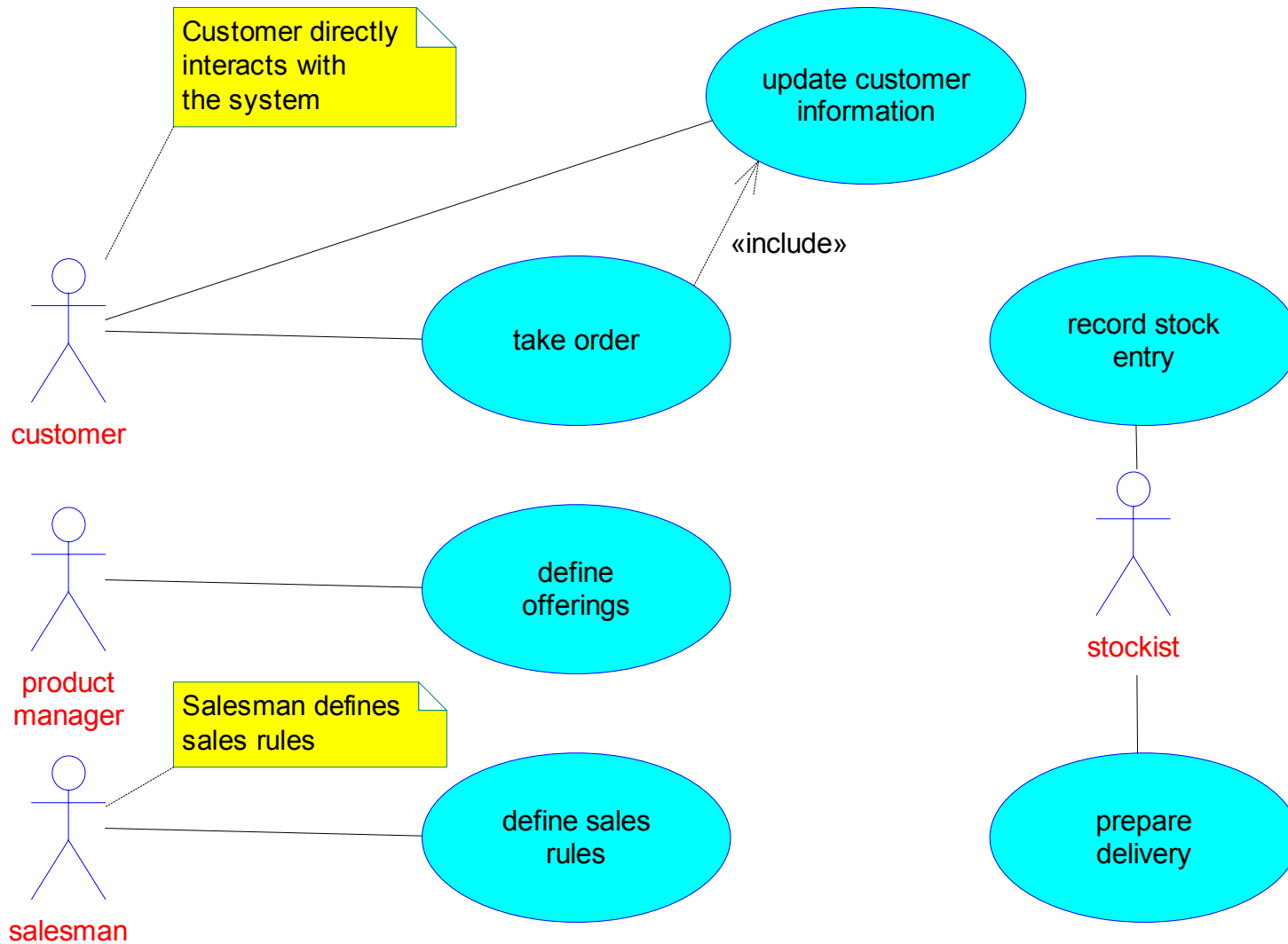
Lookup for customer
If customer not known then
 Enter details of new customer
End of customer registration
Create a new order
As often as the customer likes
 Whatever the customer likes
 Set order details
 Add good to order
 Remove good from order
 End selection
End iteration
Add discount



Types of Business Rules



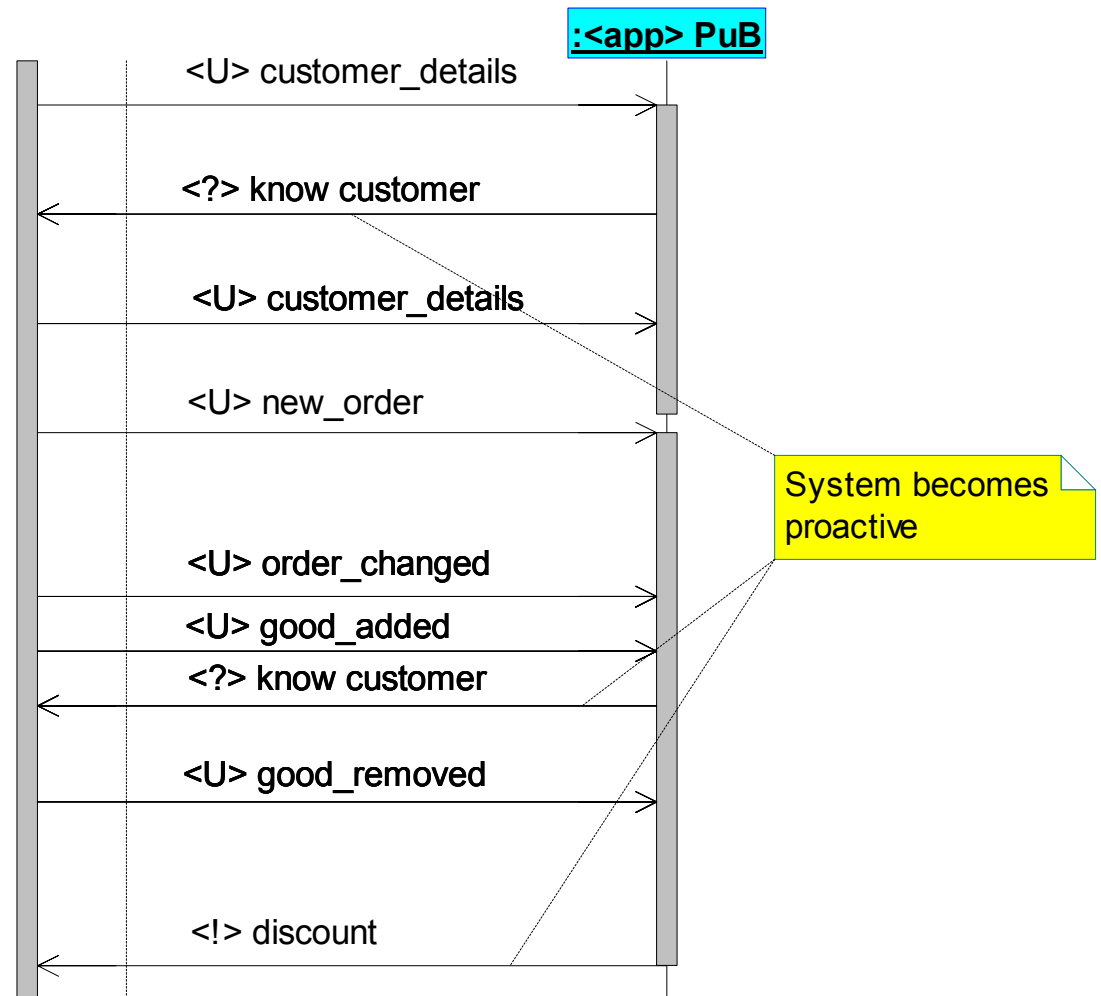
Use Case Model - BRA



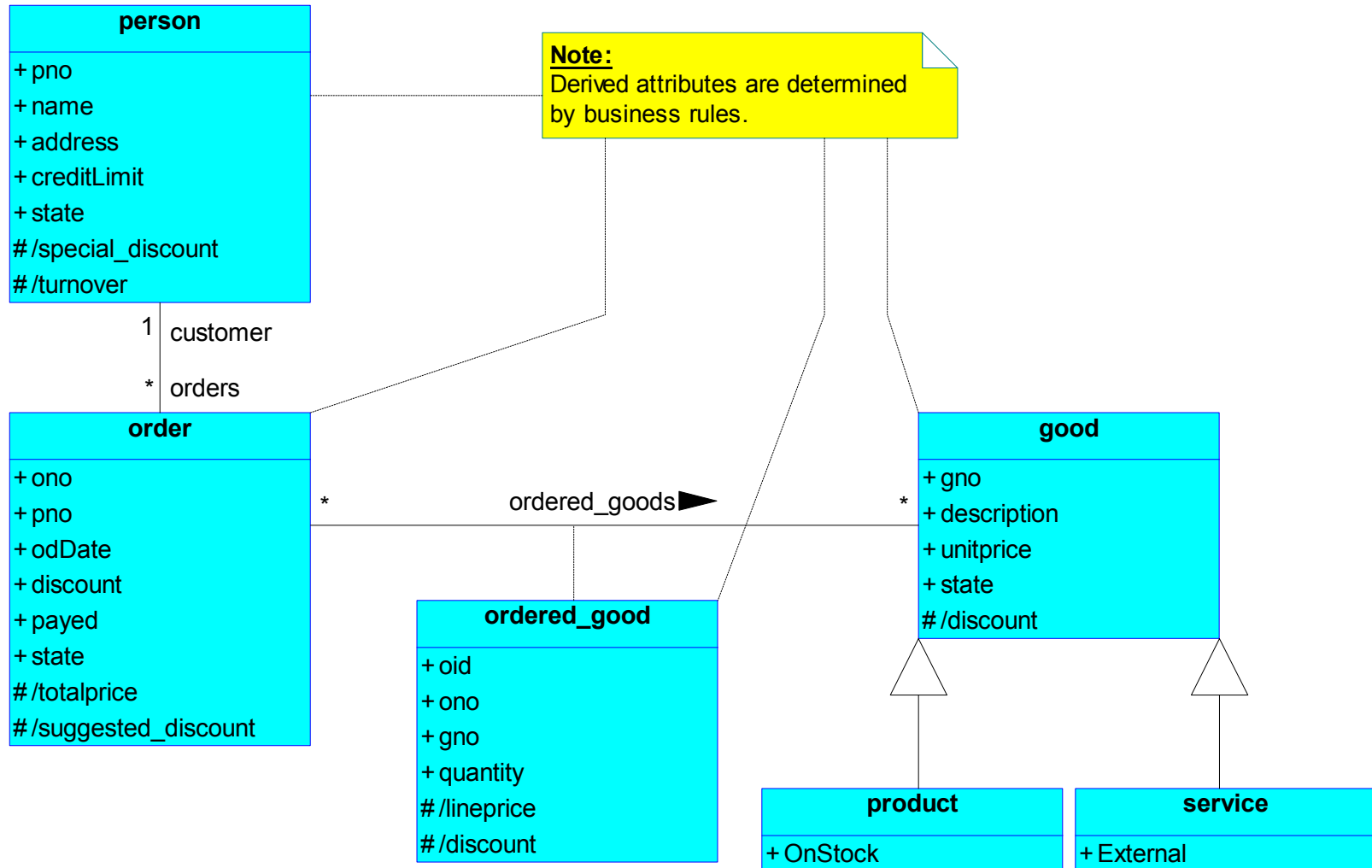
Use Case "take order" - BRA



Customer enters some details
 If system recognizes customer then
 System recognizes customer
 Else
 Customer enters some more details
 End of customer identification
 Create a new order
 As often as the customer likes
 Whatever the customer likes
 Set order details
 Add good to order
 System suggests complementary product
 Remove good from order
 End selection
 End iteration
 System determines discount



Business Objects - BRA



Some Business Rules



"Customers living in Zürich receive 10% discount, others just 5%."

```
derive if person.address = 'Zuerich' then
    person.special_discount is 10
else
    person.special_discount is 5.
```

"The product "Hürliman" gets an promotion discount of 7%."

```
derive if good.description = 'Huerlimann' then
    good.discount is 7
else
    good.discount is 0.
```

Some Business Rules (2)



"As a promotional action, customers with less than 200.- turnover will receive a discount of 5% in addition to their special discount."

```
derive if order.customer.turnover < 200 then
    order.suggested_discount is
        order.customer.special_discount + 5
else
    order.suggested_discount is
        order.customer.special_discount.
```

```
derive person.turnover is
    person.orders.ordered_goods.lineprice=>sum.
```

Business Rules are orthogonal to Objects



"Whenever the total amount of an order exceeds the credit limit of its customer, an amount of 30% has to be paid before delivering the goods."

Affects

- order
- ordered_good
- person
- good

Relevant when

- Ordering an additional good
- Increasing the good's price
- Decreasing the customer's credit limit
- All opposite events to revert to "normal state"

Designing Business Rules



The screenshot displays the ARTiSAN Real-time Studio interface. The main window, titled "Text Diagram - Business Rules", contains the following rules:

```
Initial Business Rules  
derive if person.address = "Zuerich" then  
    person./special_discount is 10.  
else  
    person./special_discount is 5.  
  
derive person./turnover is person.orders.ordered_goods./lineprice=>sum.  
  
derive order./totalprice is order.ordered_goods./lineprice=>sum - order.discount.  
  
derive if order.customer./turnover < 200 then  
    order./suggested_discount is order.customer./special_discount + 5  
else  
    order./suggested_discount is order.customer./special_discount.
```

The left sidebar shows a project tree with the following structure:

- order
 - ordered_good
 - Subclasses
 - Superclasses
 - Roles
 - Events Received
 - Events Sent
 - Interfaces
 - States
 - State Diagrams
 - /discount
 - /lineprice
 - gno
 - oid
 - ono
 - quantity
 - person
 - Subclasses
 - Superclasses
 - Roles
 - customer
 - Events Received
 - Events Sent
 - Interfaces

The bottom right shows a class diagram with two classes: **order** and **good**. The **order** class has attributes: +ono, +pno, +odDate. The **good** class has attributes: +gno, +description, +unitprice. A relationship named "ordered_goods" connects the **order** class to the **good** class. The **order** class has a multiplicity of "*" and the **good** class has a multiplicity of "*".



- **Application = Business Objects + Business Rules (+ GUI)**
- **Business Rules** are shared assets of an enterprise as **Business Objects** (should be)
- **Business Rules** should be maintainable by **Business People**
- **WHAT** not **HOW!**

Document History



Date	Author	Modification
20.09.2001	M. Schacher	Initial version