ENTERPRISE MODELING AND REFINEMENT IN UML

Branislav Lazarević, Siniša Nešković
(branko & sinisa @fon.fon.bg.ac.yu)
Faculty of Organizational Sciences,
University of Belgrade
Jove Ilica 154, Belgrade, Yugoslavia

UML AND REFINEMENT

- Stepwise refinement was fundamental and very useful abstraction in program and model developments since Hoare's *Structured Programming* till Structured Analysis and Design and other similar structured development methodologies.

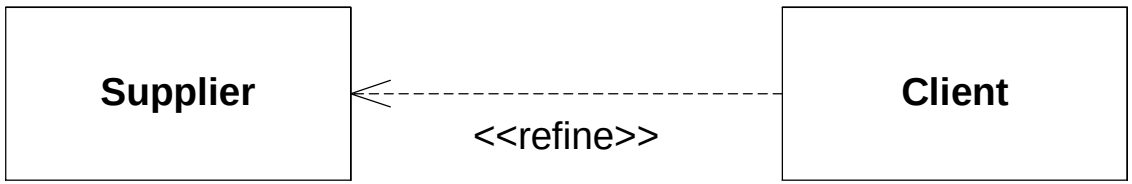
- Stepwise refinement means some kind of multilayer abstractions in modeling, i.e. some kind of decomposition in modeling.

- Object-oriented approaches do not use stepwise refinement extensively what is generally recognized as serious drawback in enterprise modeling.

- The presentations deals with the problem of behavior refinement in UML.

REFINEMENT IN UML?

1. ***Refinement relationship -*** "represents a fuller specification of something that has already been specified at certain level of detail or at a different semantics level". It may contain a formal description of mapping, but not necessarily.



2. Refinement relationship is defined very generally. It seems that additional semantics can be given only within a specific software development process (methodology).

3. What can be Supplier and what can be Client depends on a development process

REFINEMENT SEMANTICS

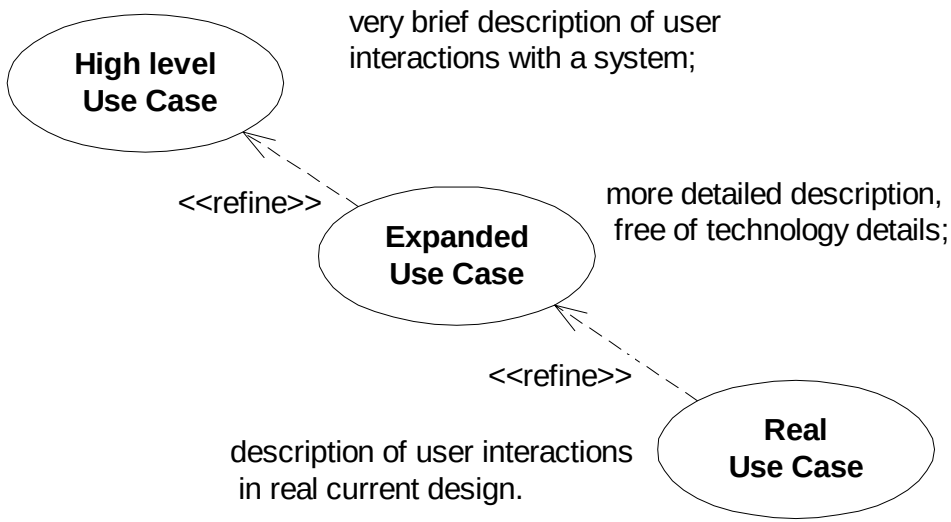
- Early attempts to incorporate refinement in OO development where directed toward different ways of combination of structured and OO development – most often, so called, FOO approach (Functional analysis, Object design and Object implementation) [Constantine 1991].

- However the dogma that everything (Analysis, Design, Implementation) have to be OO, nothing structured, have won.

  Use Cases has become functional model in UML.

USE CASES AND REFINEMENT SEMANTICS

- Different kinds of use cases to incorporate different level of abstractions of requirement specification [Larman 1998]



- Use Case dependencies << extend>>, <<include>> can be treated as refinement relationship, too

REFINEMENT

The Catalysys approach [DSouza 1999] suggested very interesting and important way to incorporate refinement in UML:

- Promote the action (“everything that happens: an event, task, job, message, change state, interaction or activity”) as the “first class entity” in the model);
- The core of a model should be very small, just objects and actions;
- Introduce refinement relationship as a relationship between a particular design and its specification, with mapping between them and justifications of design choices;
- Four kinds of refinement: action refinement, object refinement, operation refinement, model refinement

REFINEMENT SEMANTICS

Refinement semantics is here developed having in mind the "System-Theoretic Modeling Lifecycle" (STML) [Calman 1954] consisting of the following stages:

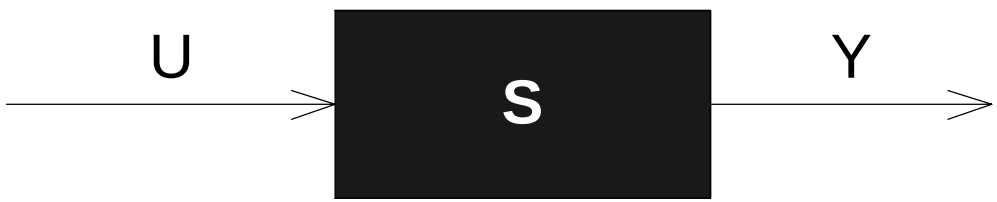
1. ***Identification***, system model is given as a set of functions representing input-output transformation
2. ***Realization*** in which one first choose a model (a theory) to realize the system and then tries to find a minimal or satisfactory realization of identified set of functions within the chosen model.

   Since realization may contain functions, these two steps can be iterated until primitive realization is reached.
3. ***Implementation,*** transformation of the primitive realization model into a given implementation environment.

1. IDENTIFICATION (SPECIFICATION)

Identification (Specification), a system model is given as a collection of functions representing input-output transformation;

$$\mathbf{S:\{F:\ U \dashrightarrow Y\}}$$

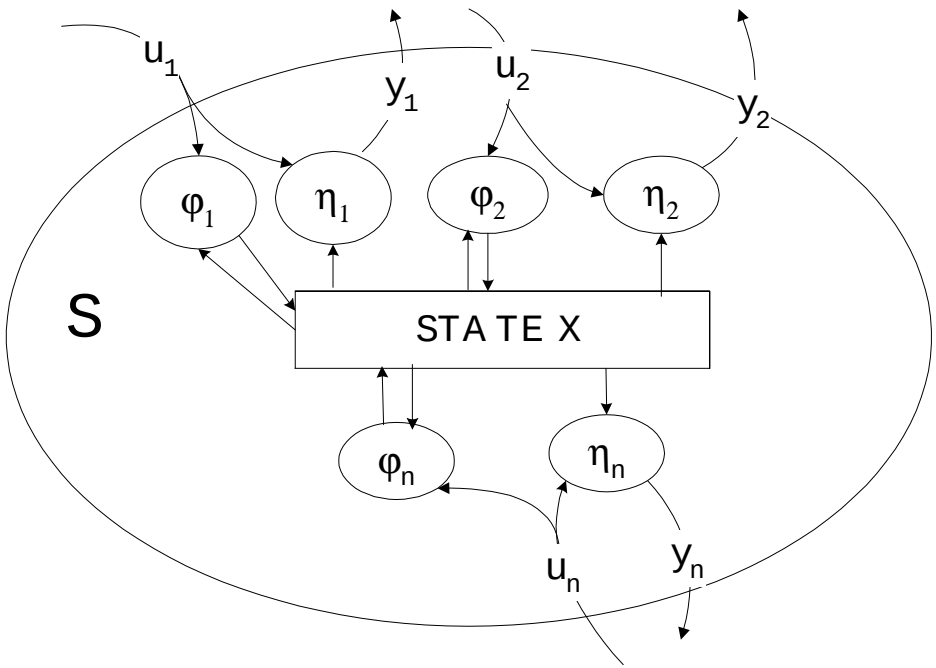


Any kind of system analysis will do if provides proper set of functions.

2. STATE SPACE REALIZATION

$$\varphi: U \times X \rightarrow X; \eta = U \times X \rightarrow Y$$

One-step state transition function and output transformation - transaction in respect to State X



Minimal realization

Minimal coupling between functions

If state implementation is relational database, minimal realization means normalization

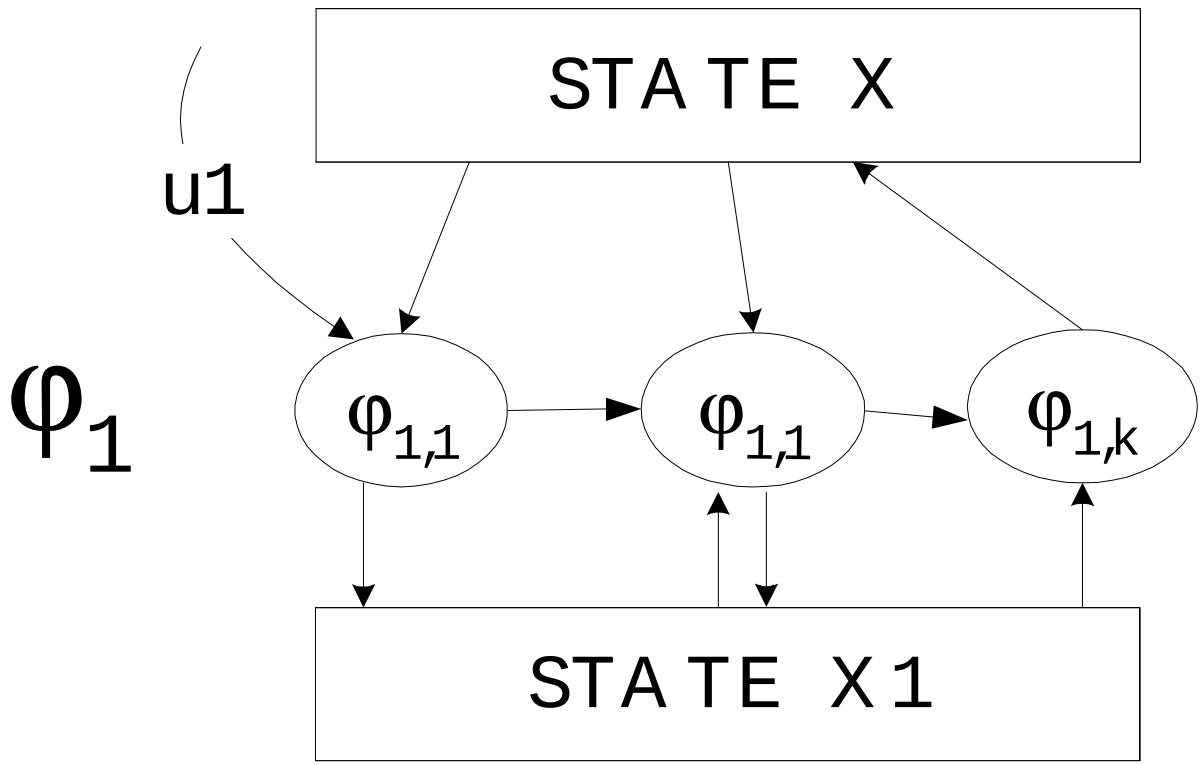
STATE SPACE REALIZATION

The terms function is defined as an "atomic computation" that results in a change in the state of the model or the return of a value". The terms function, operation and action are treated here as synonyms.

They can be also treated as transactions (logical units of work), with ACID properties at specified level of abstractions. One can consider his/her requirement to be fulfilled as an function although it is realized as a business process, or an interactive application (Use Case).

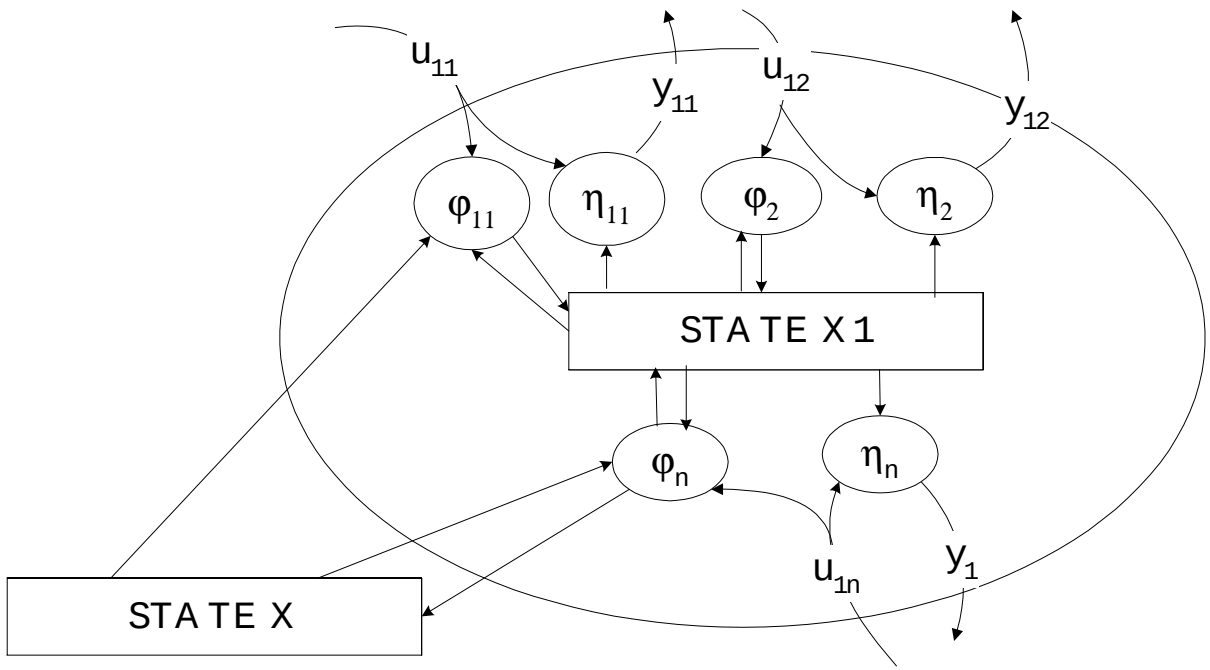
FURTHER REALIZATION

New state X1 is introduced to resolve computational complexity of φ_1. Function φ_1 is realized as a program (method, or collaboration)

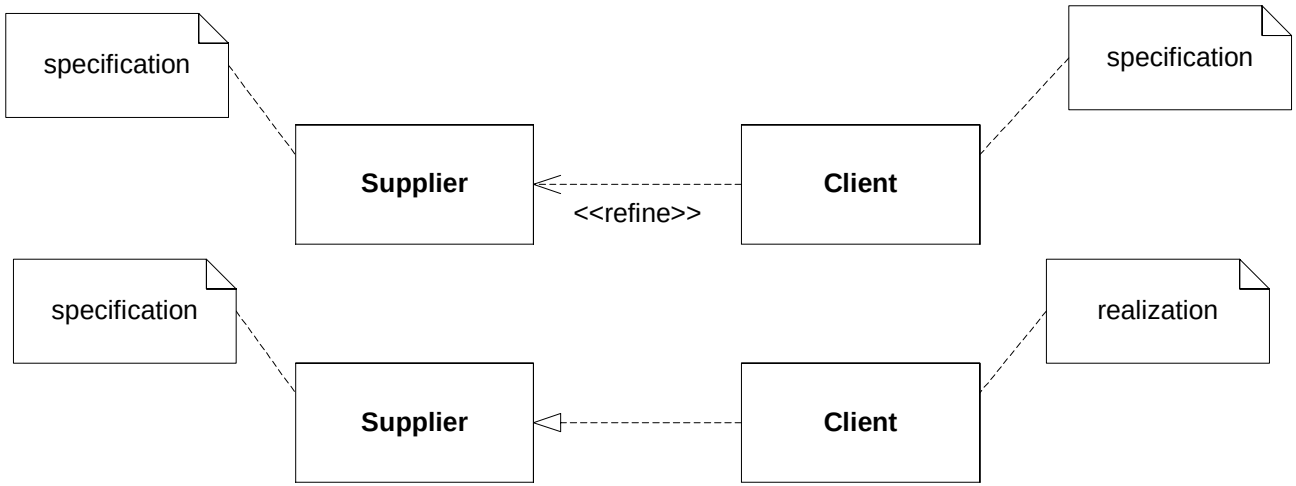


FURTHER REALIZATION

New state X1 is introduced to resolve both computational complexity of φ_1 and structure complexity of the input u_1. Subfunction is used to construct u1 in the state X1. Function φ_1 is realized as a "use case" (an "interactive application") or business process.



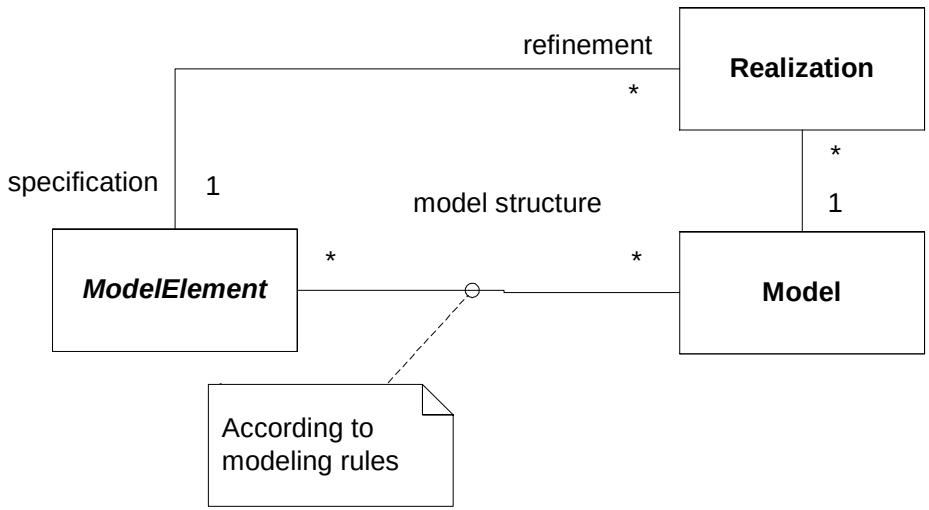
REALIZATION AND REFINEMENT



- What is realization on one level of abstraction can be consider as specification on the other one.

- Realization relationship should be the main mechanism to specify refinement since ***"realization is an indication of the inheritance of behavior without the inheritance of structure"***

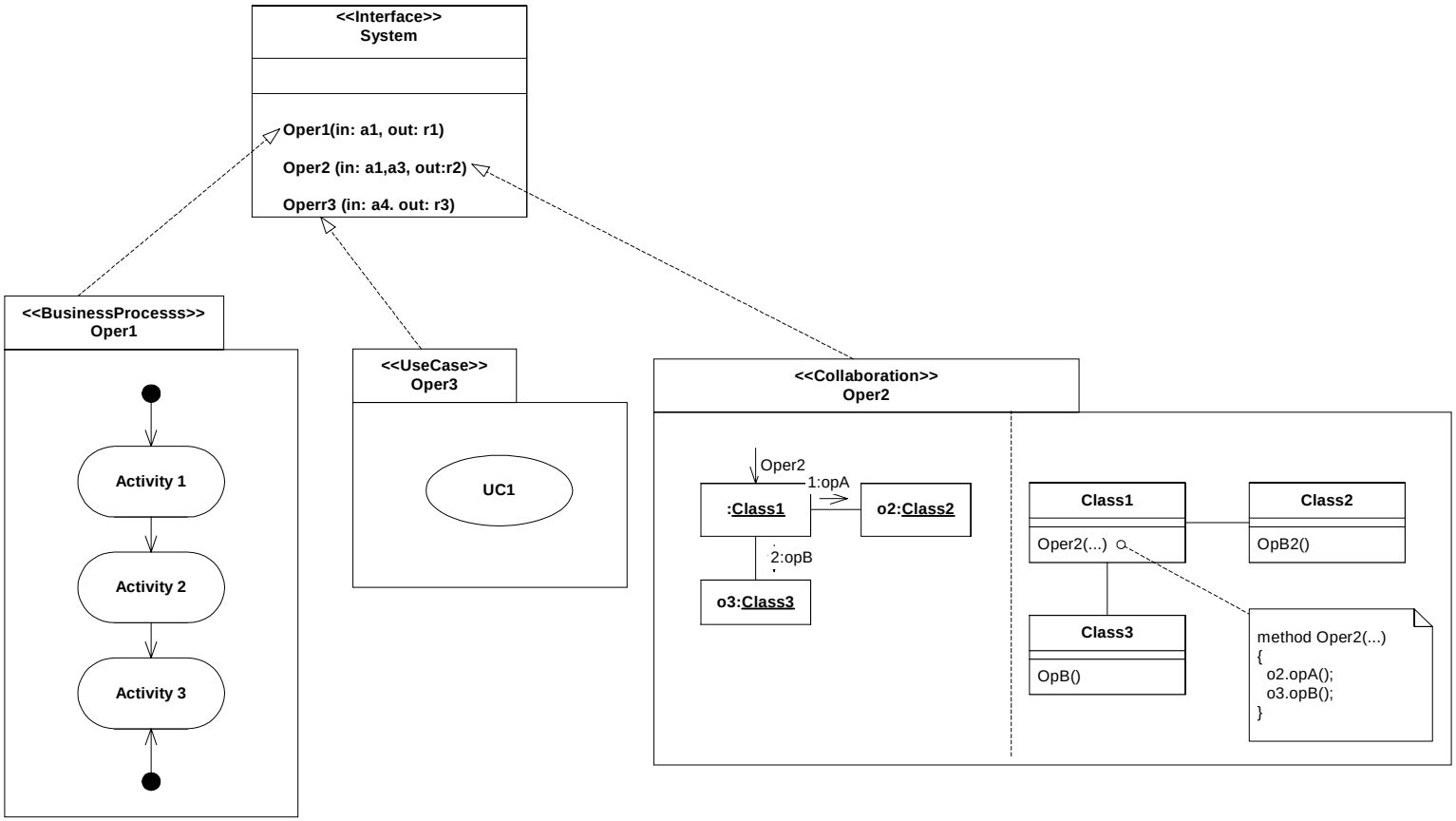
SEMANTICS OF STEPWISE REFINEMENT THROUGH REALIZATION



- UML supports these relationships at higher level of abstraction
- However, the semantics of realization should relate model element and model that realize it at lower level of abstraction
- Model is constructed of model elements according to specific modeling rules

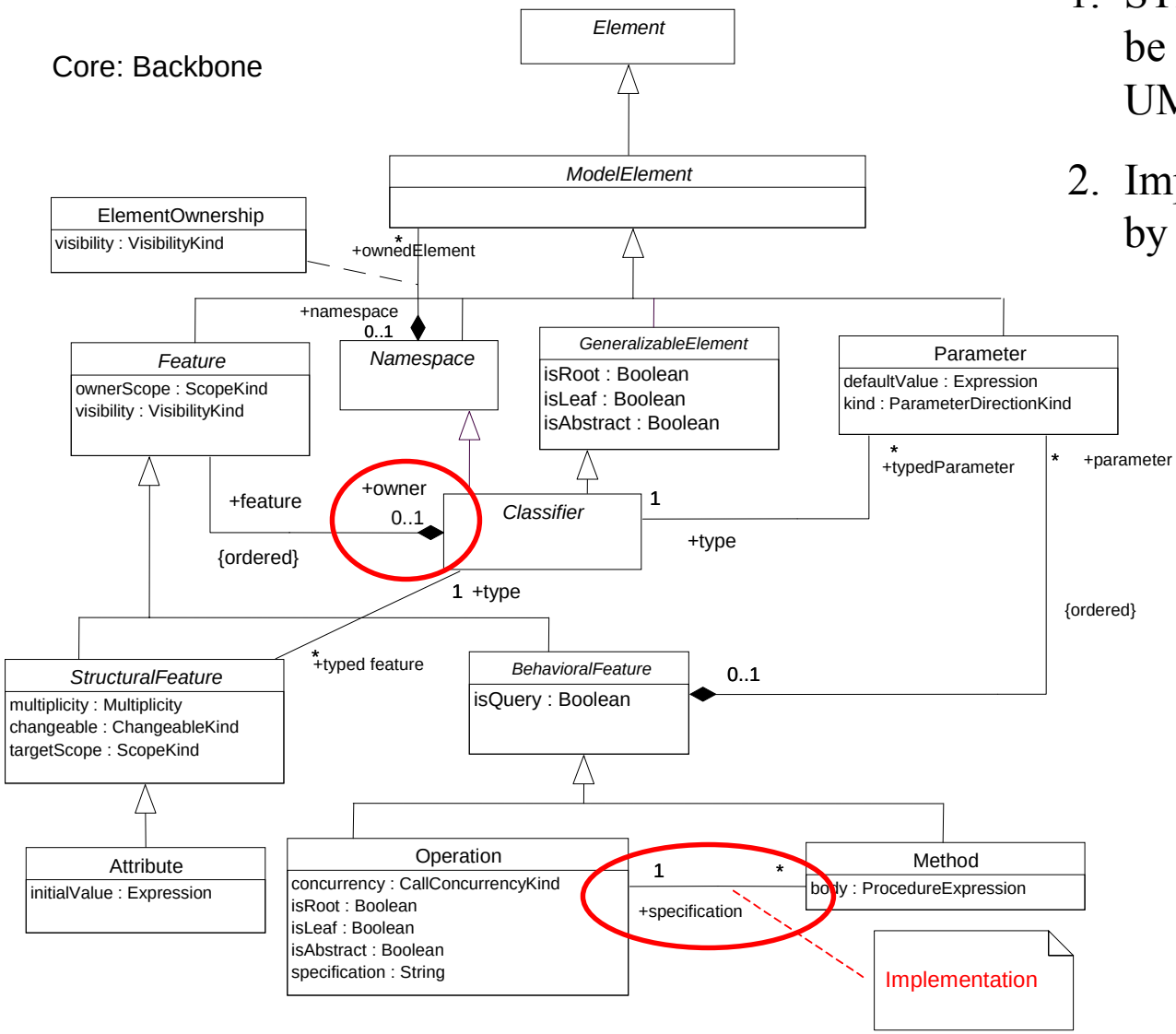
REALIZATION OF OPERATION

- STLM implies stepwise refinement through realizations of functions (operations)



UML metamodel

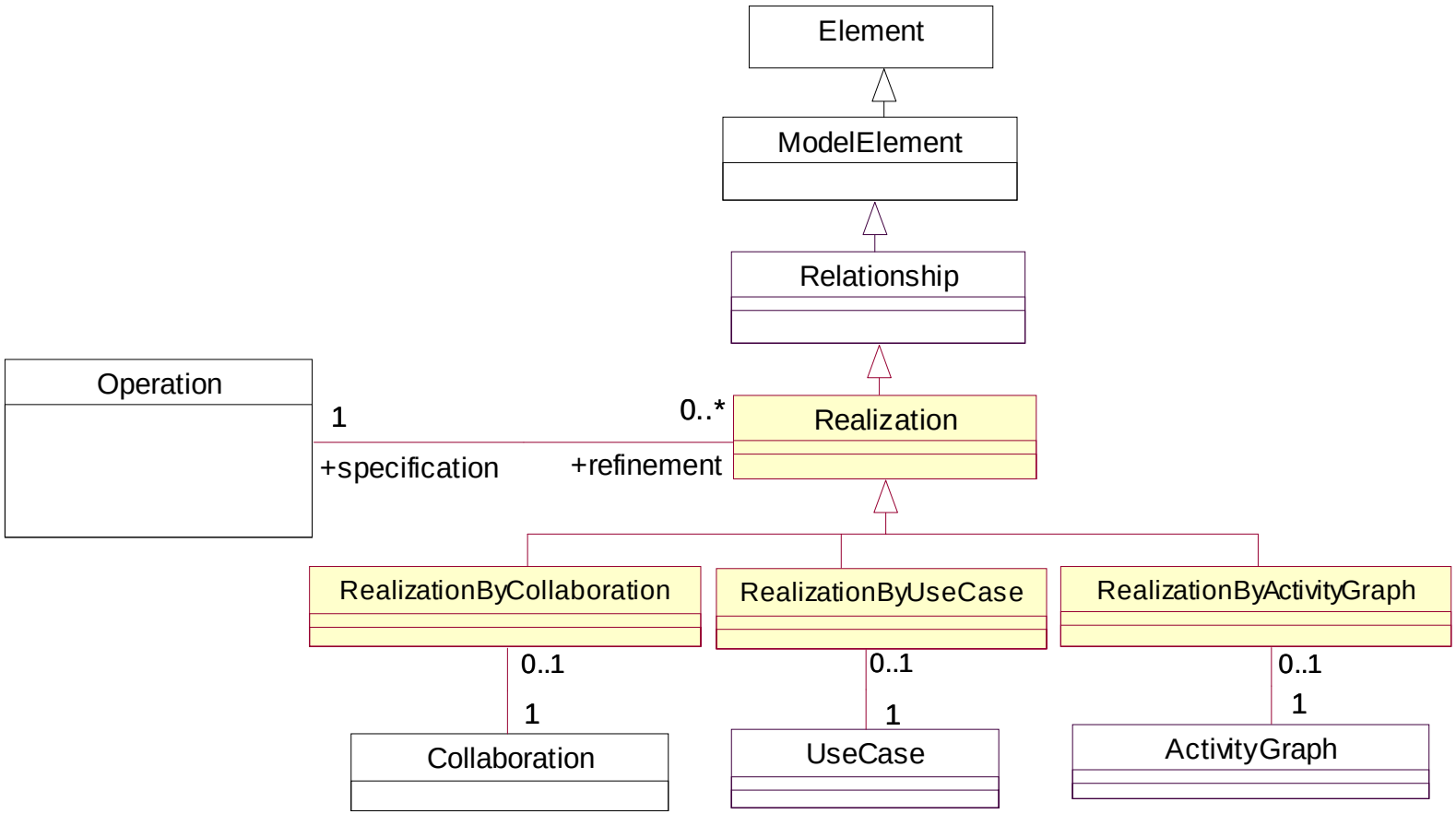
Core: Backbone



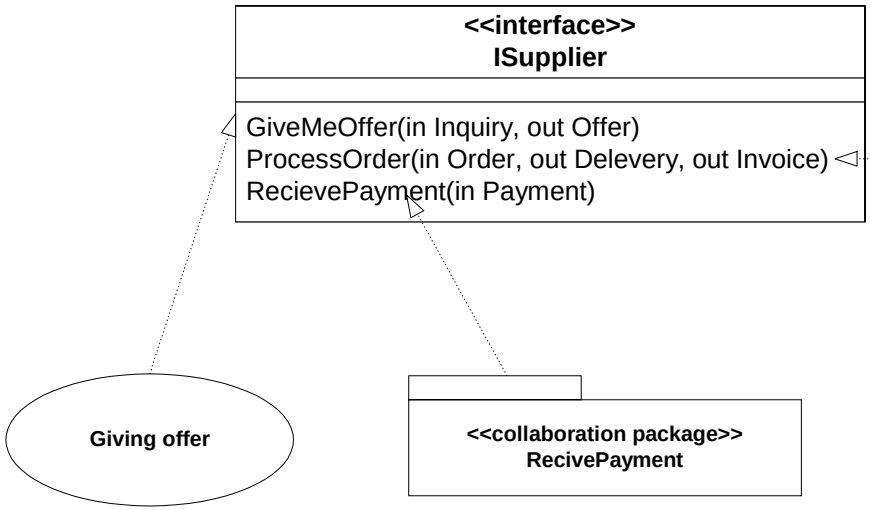
1. STML requires operations to be “first class entities” what UML allows
2. Implementation of operations by method already exists

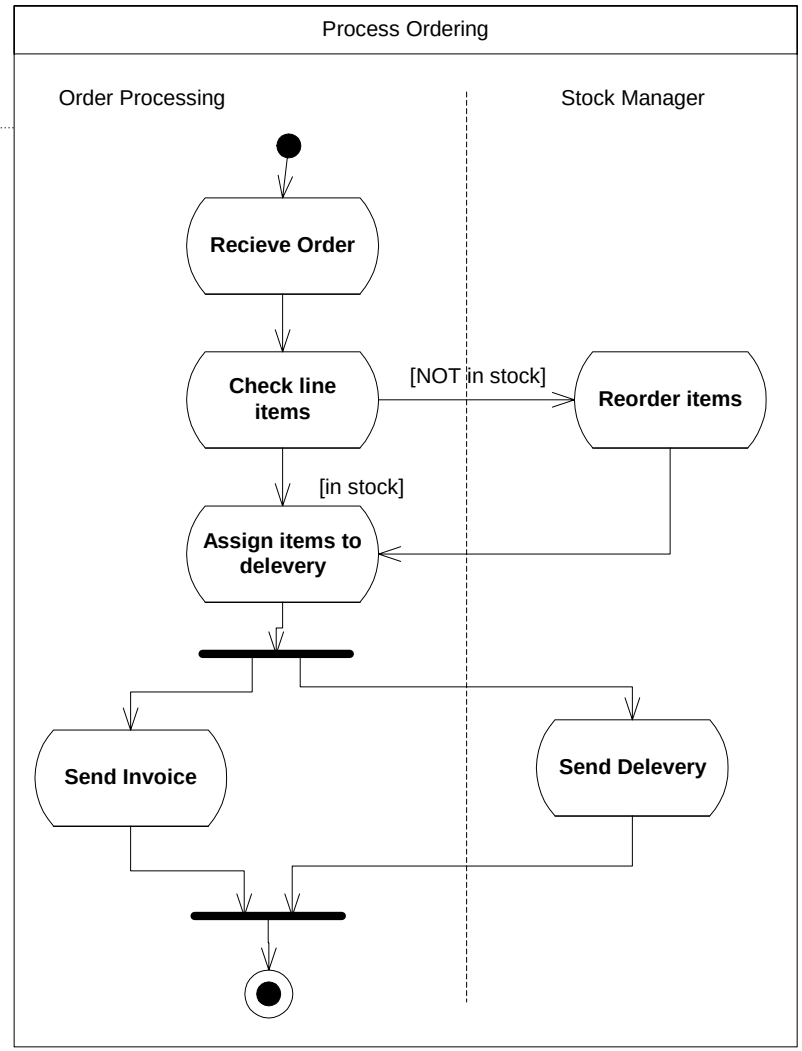
Extensions of UML metamodel

Since STML is based on operation realization, the following of UML metamodel is given:

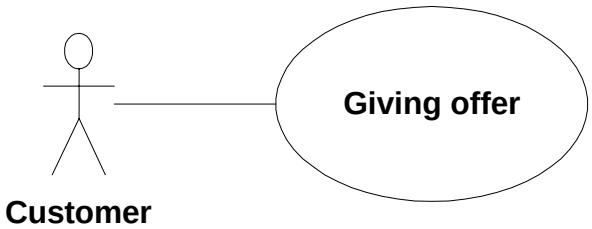


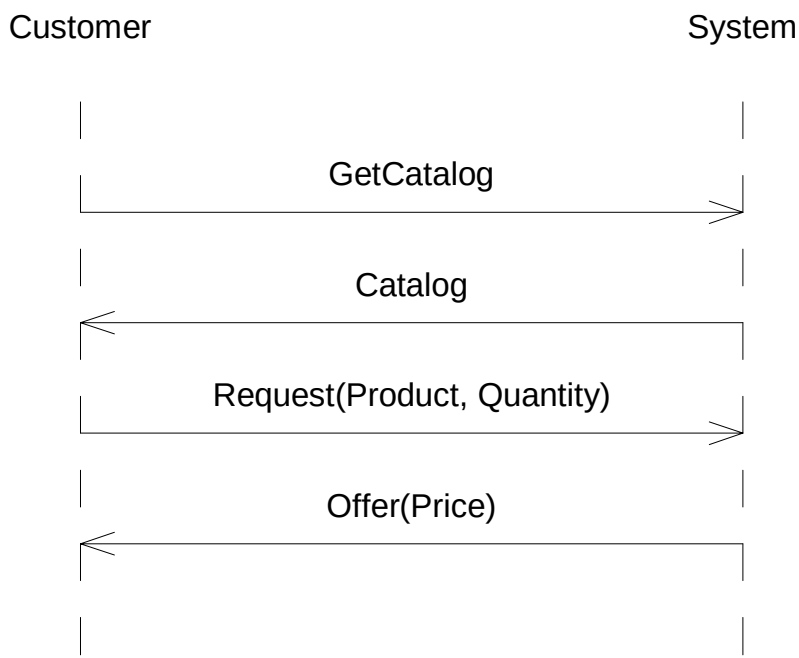
Example


<<interface>>
ISupplier
GiveMeOffer(in Inquiry, out Offer)
ProcessOrder(in Order, out Delevery, out Invoice)
RecievePayment(in Payment)
Giving offer
<<collaboration package>>
RecivePayment

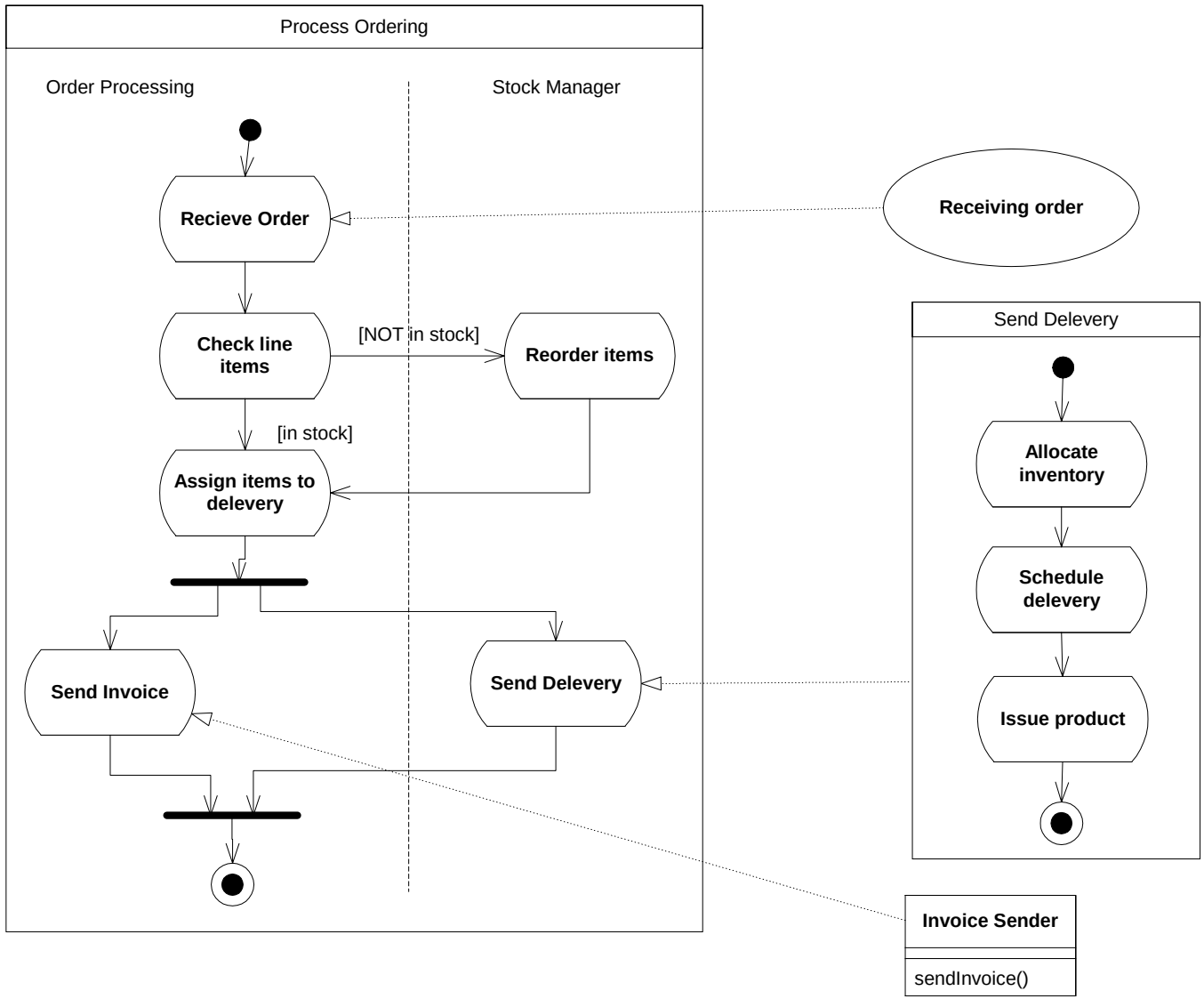

Process Ordering
Order Processing
Stock Manager
Recieve Order
Check line items
[NOT in stock]
Reorder items
[in stock]
Assign items to delevery
Send Invoice
Send Delevery

Operation realization by Use Case

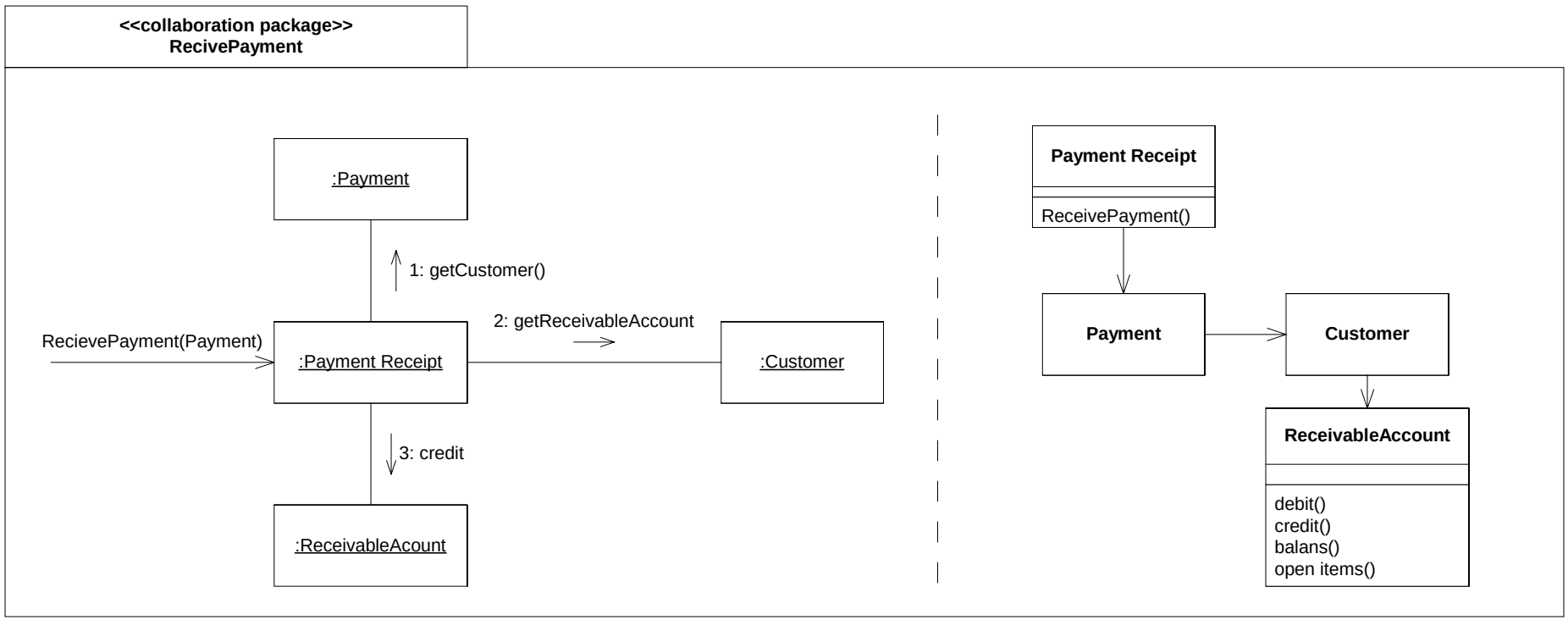




Operation realization by business process



Operation realization by program



Conclusion

1. Detailed semantics of refinement can be given only within a specific software development process (methodology).

2. Refinement in STML is given through realization of functions (operations) within chosen model. If model contains operations as model elements, refinement can continue.

3. Implementation is association between operation and method.

4. Semantics of operation realization is represented as an extension of UML metamodel

References

[Constantine 1991] Constantine, L.L., Hebderson-Sellers, B. *Object-oriented Development and Functional Decomposition*, JOOP, January 1991, pp 11-16

[D'Souza 1999] D.F. D'Souza, and A.C. Wills. 1999. *Objects, Components and Frameworks with UML – The Catalysis Approach*, Addison-Wesley Longman

[Kalman 1954] Kalman, R.E., Falb, P.L., Arbib, M.A., 1969 . *Topics in Mathematical Systems Theory*, McGraw-Hill

[Larman 1998] Larman, C., 1998. *Applying UML and Patterns*, Prentice Hall