



# Лекция 9

## Основни концепции на ориентирания към сценарии изглед на системата

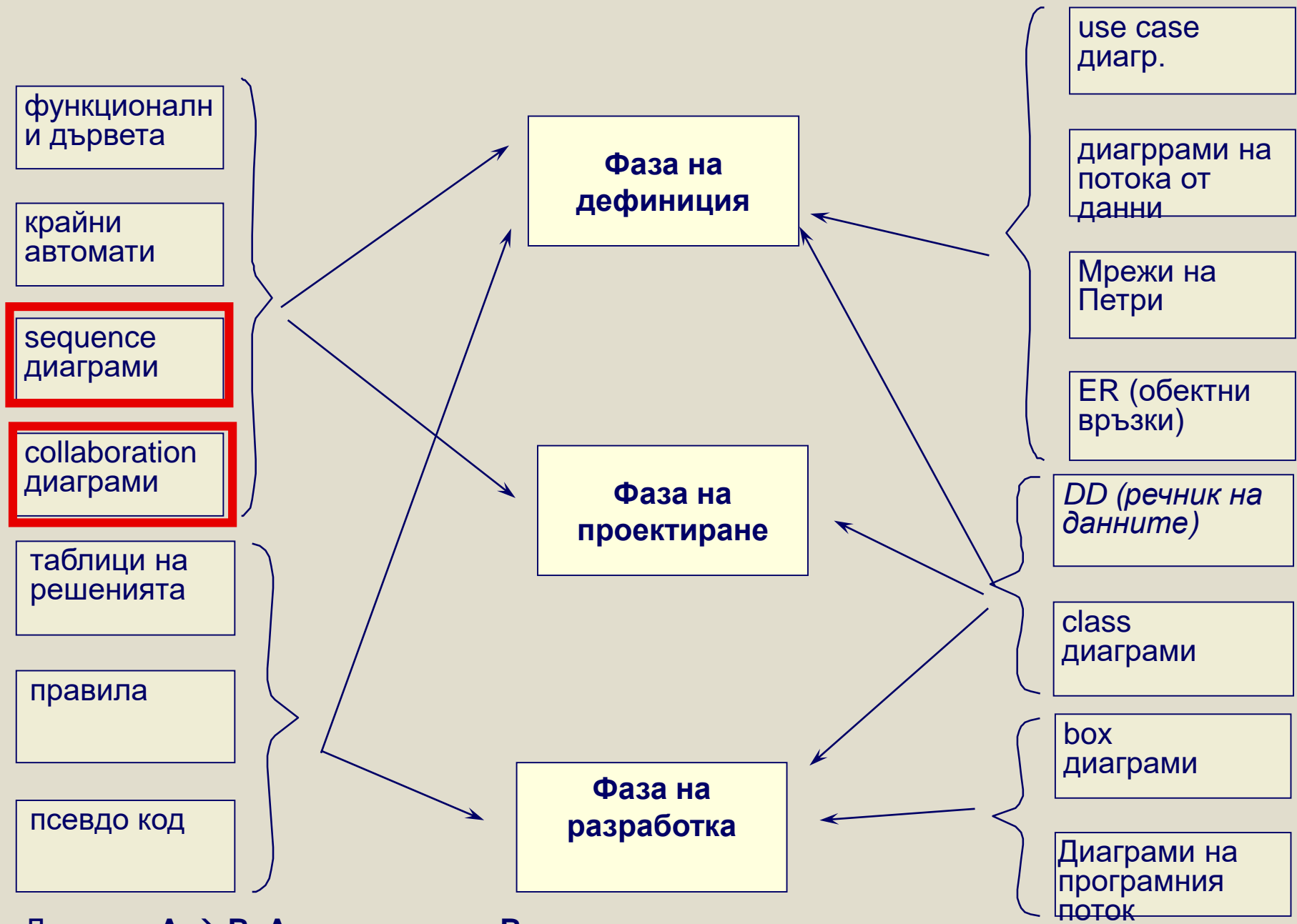
DAAD Project  
“Joint Course on Software Engineering”

Humboldt University Berlin, University of Novi Sad, University of Plovdiv,  
University of Skopje, University of Belgrade, University of Niš, University of Kragujevac

Parts of this topic use material from the textbook  
H. Balzert, “Software-Technik”, Vol. 1, 2nd ed., Spektrum Akademischer Verlag, 2001

# Основни концепции: ориентиран към сценарии изглед на системата

|   |                       |                        |                             |                                |                         |                    |                              |                        |                       |                        |                |                       |                    |
|---|-----------------------|------------------------|-----------------------------|--------------------------------|-------------------------|--------------------|------------------------------|------------------------|-----------------------|------------------------|----------------|-----------------------|--------------------|
| <div>Concepts and Views</div> <div>Alternative Notations</div> <div>Often used</div> <div>Rarely used</div> |                       |                        |                             |                                |                         |                    |                              |                        |                       |                        |                |                       |                    |
| Function tree   | Use Case Diagram 1987 | Data flow diagram 1966 | <i>Data-Dictionary</i> 1979 | Entity Relationship Model 1976 | Class diagram 1980/1990 | Pseudo code        | Program structure chart 1973 | Program flowchart 1966 | Decision tables 1957  | Activity diagram 1997  | Petri-Net 1962 | Collaboration diagram | Sequ. diagram 1987 |
| Functional hierarchy  | Business process      | Information Flow       | Data Structures             | Entity types and relations     | Class structures        | Control structures | If-Then structures           | Finite State Automat   | Concurrent structures | Interaction structures |                |                       |                    |
| Functional View   |                       |                        | Data Oriented View          |                                | Object Oriented View    | Algorithmic View   | Rule Based View              | State Oriented View    |                       | Scenario Based View    |                |                       |                    |
|   |                       | ✓                      |                             | ✓                              |                         |                    | ✓                            |                        | ✓                     |                        |                |                       |                    |



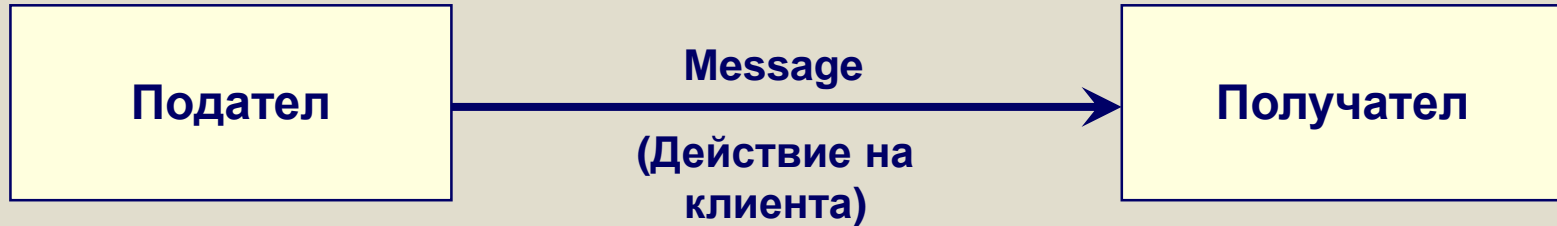
Легенда: **A → B: A се използва в B**



## 12. Основни концепции на ориентирания към сценарии изглед на системата

- a) Съобщения и сценарии
- b) Collaboration диаграми
- c) Sequence диаграми

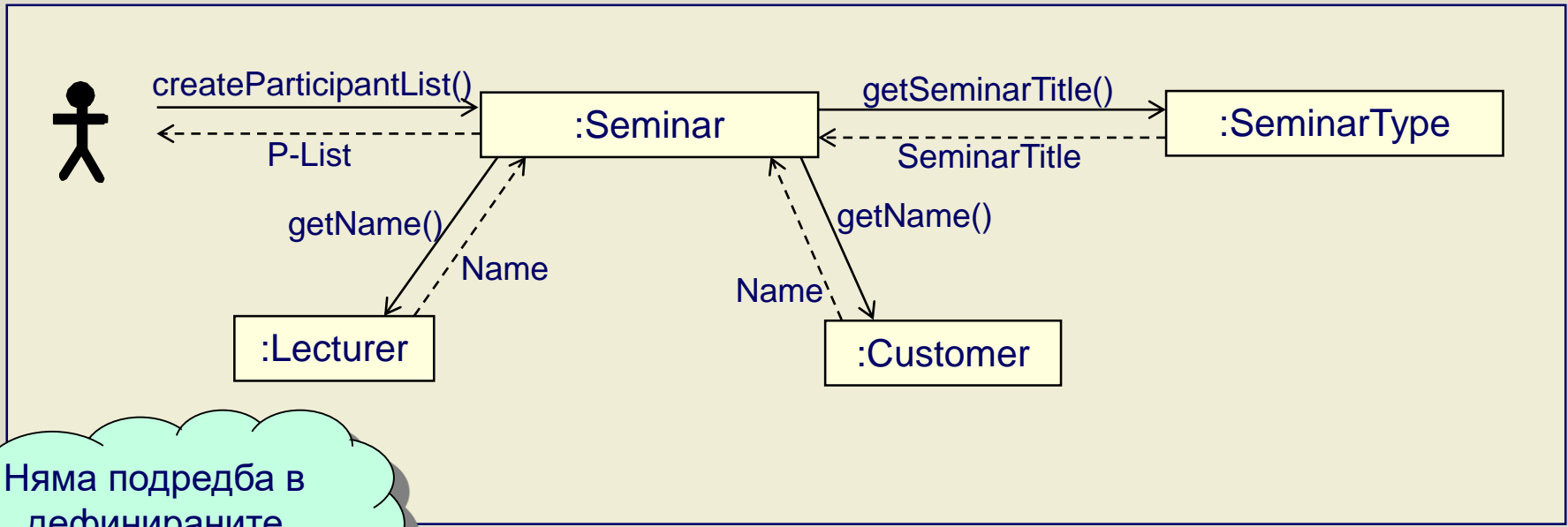
# Съобщения



## ► Съобщение

- Подателят изисква от получателя да изпълни определена услуга като изпраща съобщение
- ... като извиква операция на получателя

# Изпращане на съобщение: createParticipantsList()



Няма подредба в  
дефинираните  
съобщения

## Съдържана информация:

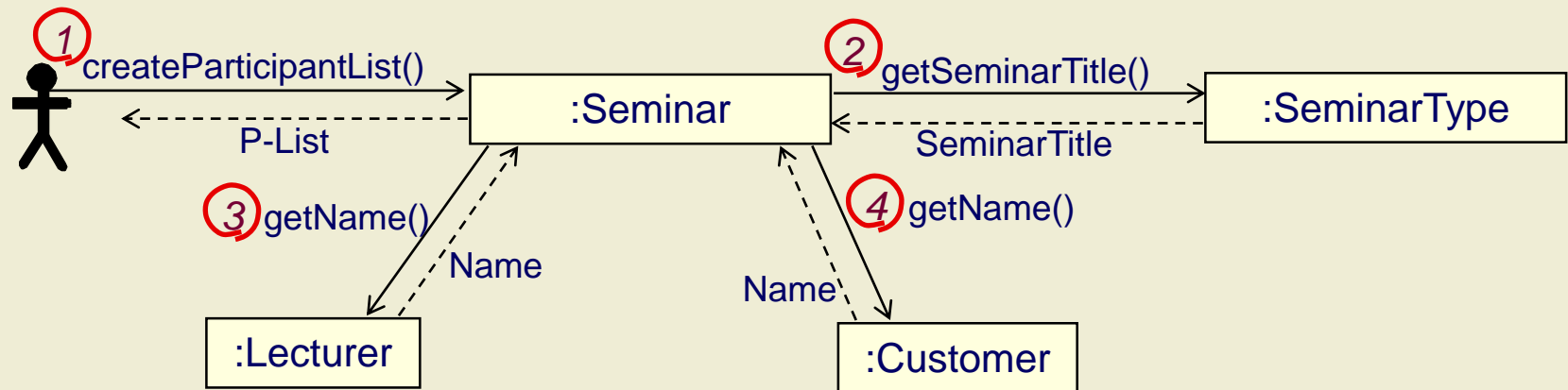
- ▶ Подател: seminar manager, seminar
- ▶ Получател: seminar, seminarytype, lecturer, customer
- ▶ 4 съобщения: `createParticipantsList()`, `getSeminarTitle()`, `getName()`, `getName()`

# Сценарии

- ▶ Описва динамична последователност от операции на системата
- ▶ Сценарий = Последователност от съобщения  
→ подредбата на съобщенията е дефинирана
- ▶ Use case може да бъде документиран чрез множество от сценарии



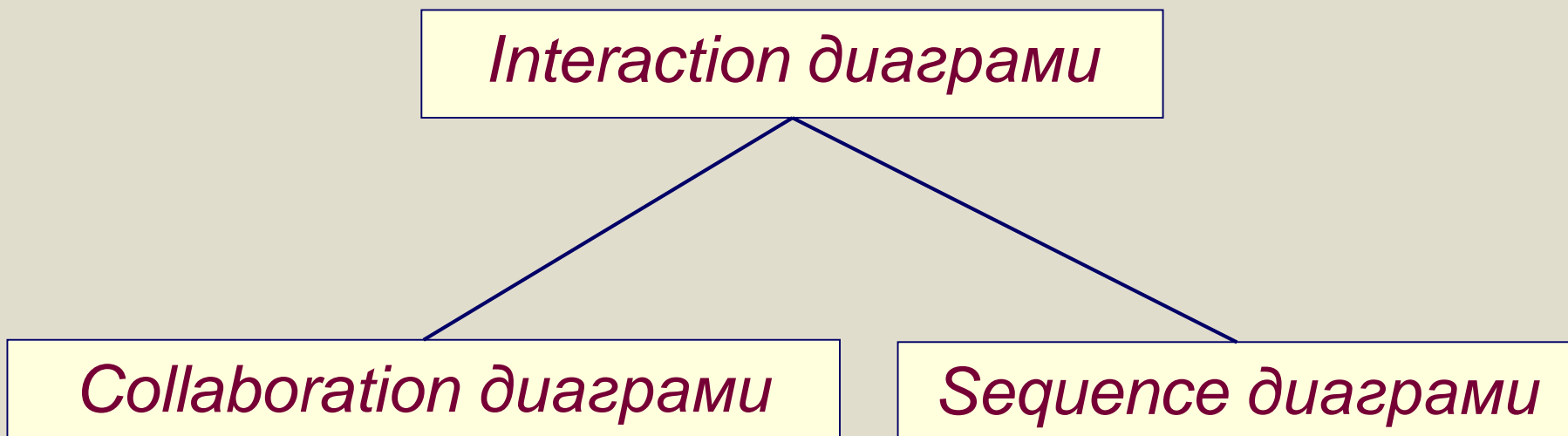
# Сценария по време на изпълнението на createParticipantList()



## 12. Основни концепции на ориентирания към сценарии изглед на системата

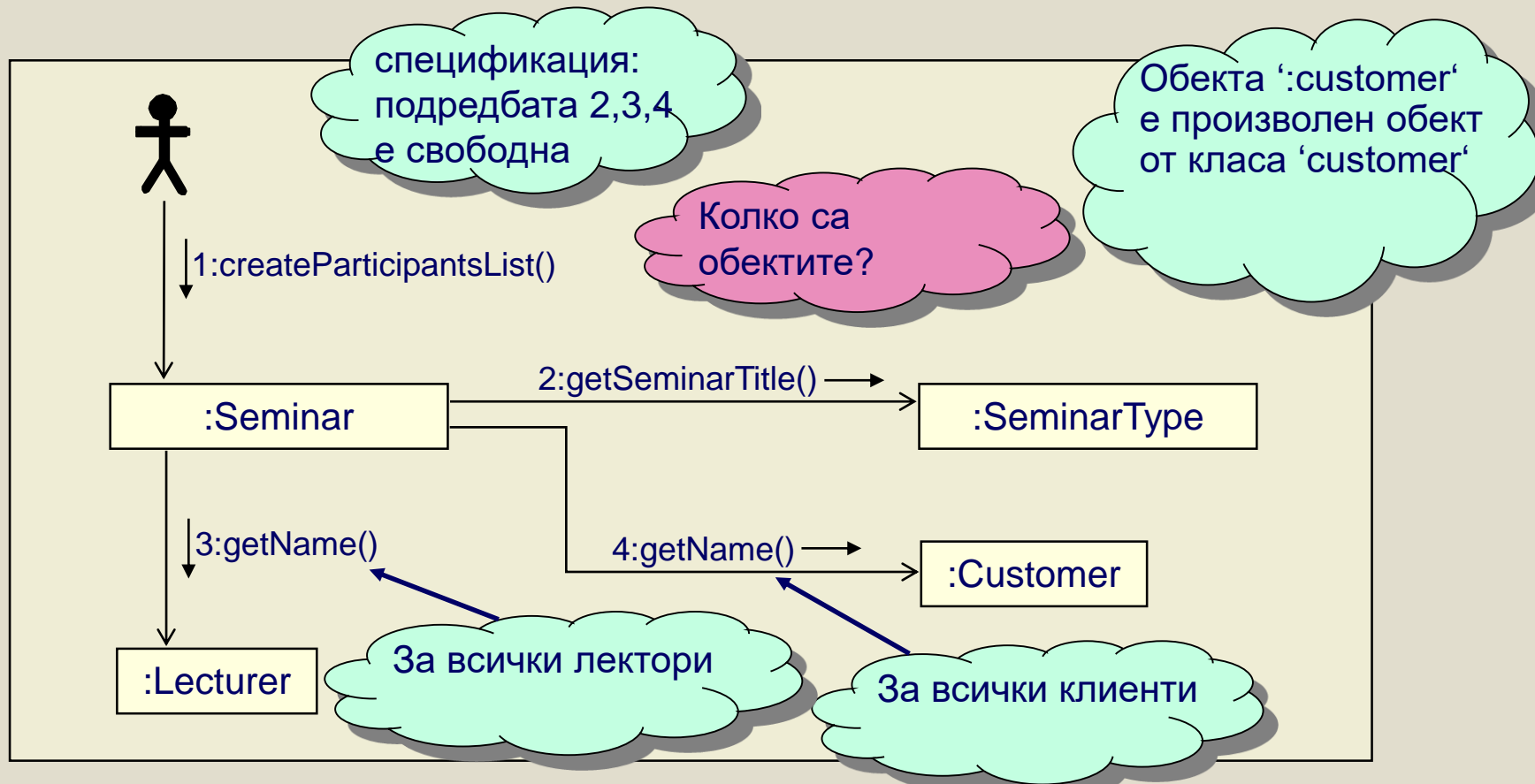
- a) Съобщения и сценарии
- b) Collaboration диаграми
- c) Sequence диаграми

# UML представяне на сценариите



Основни концепции на ориентирания към сценарии изглед на системата :  
описва последователност от съобщения между актьори и обекти на системата

# Collaboration диаграми: createParticipantList()



- Изпълнението на операцията `'createParticipantList()'` води до взаимодействие между обектите (на 4 различни класа и един актьор)
- Подредбата се осъществява с номера

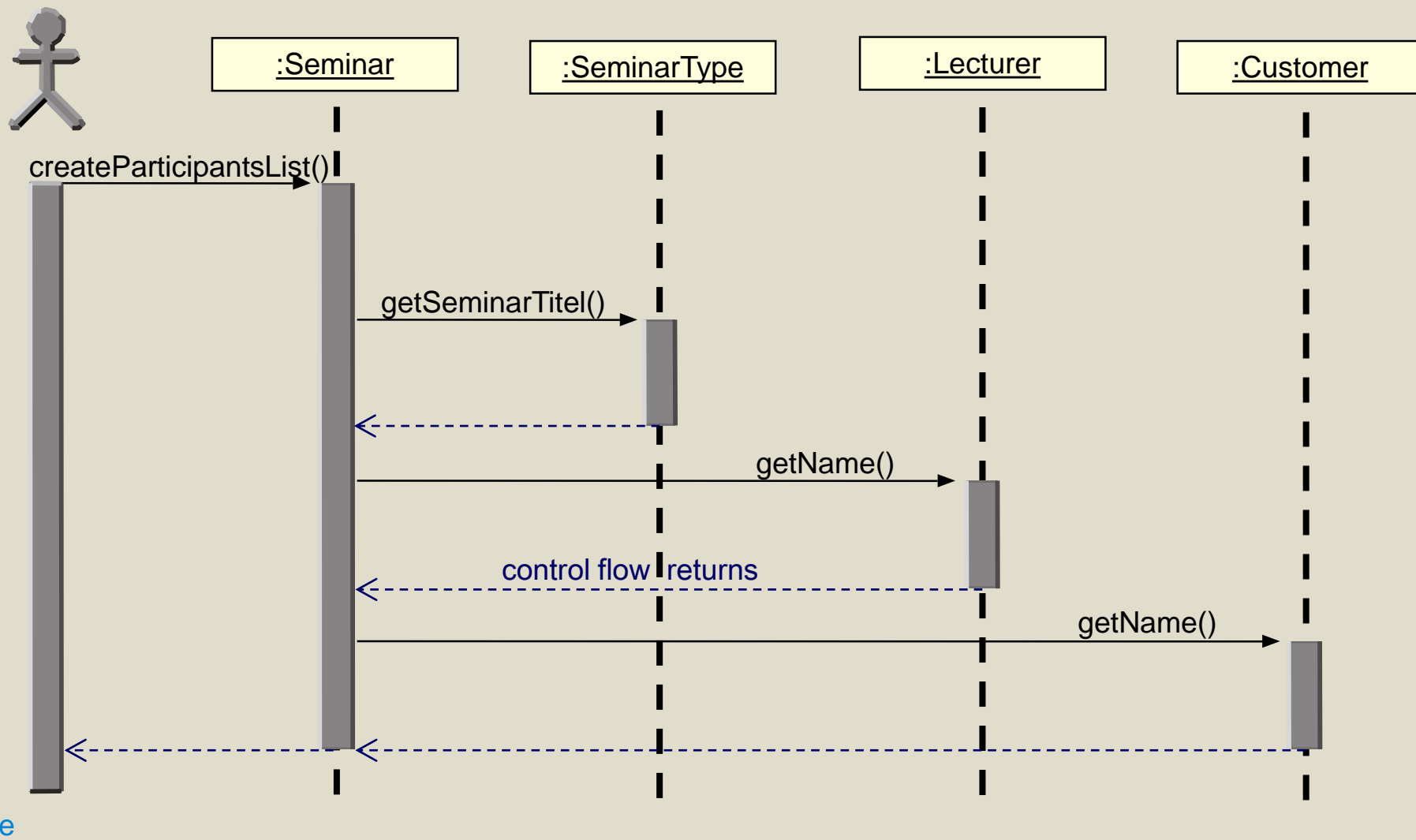
# Collaboration диаграми: живота на обектите



## 12. Основни концепции на ориентирания към сценарии изглед на системата

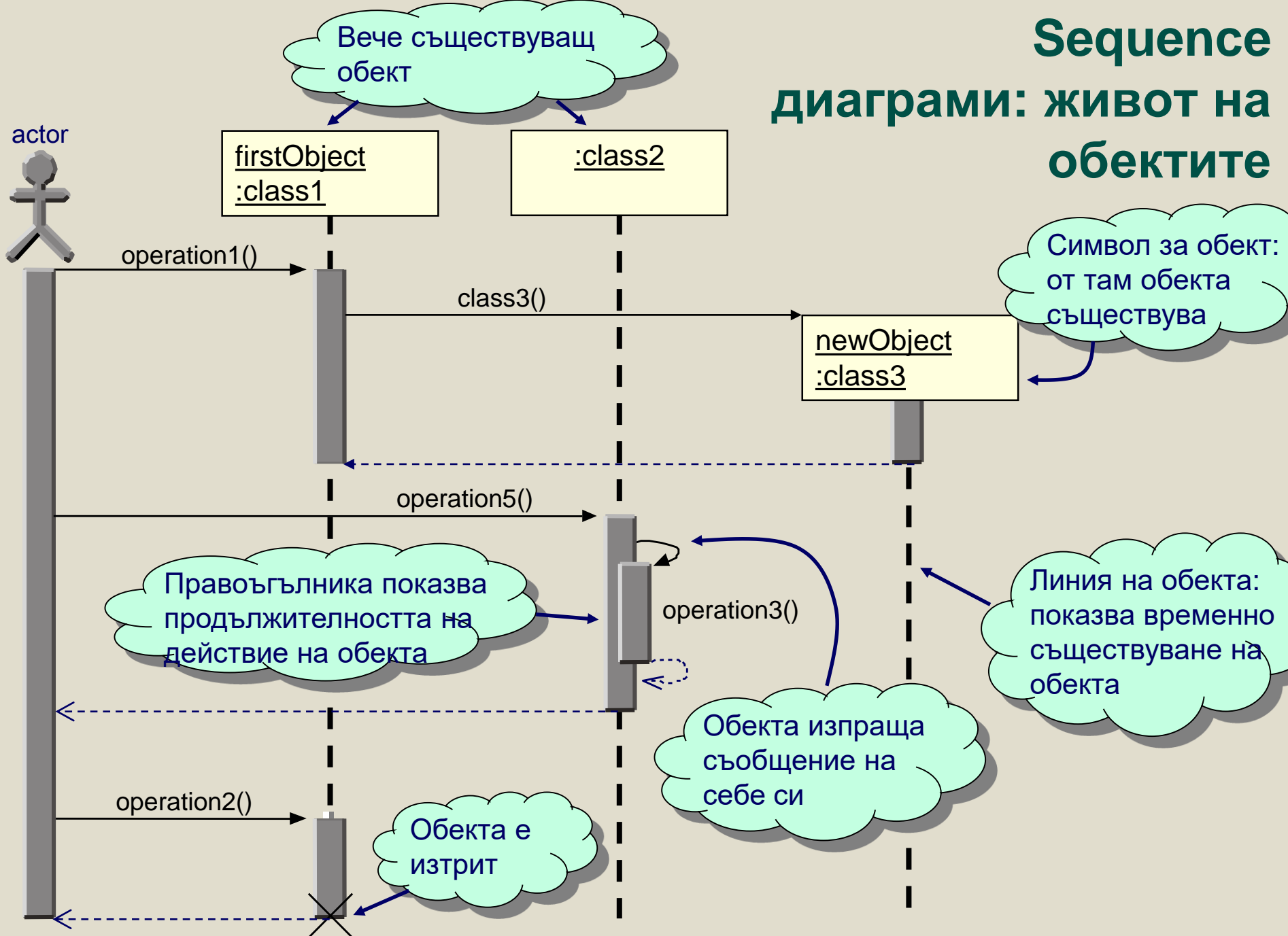
- a) Съобщения и сценарии
- b) Collaboration диаграми
- c) Sequence диаграми

# Sequence диаграми: 'createParticipantList()'



Подредени във времето

# Sequence диаграми: живот на обектите





# История

- ▶ *Dr. Ivar Hjalmar Jacobson*

- \*2.9.1939 in Ystad, Schweden

- Заместник президент на Rational Software Corporation

- ▶ Инициатор на разработката на обектно-ориентиран софтуер

- (use case направляван подход)*

- ▶ Въвежда interaction диаграмите в разработката на обектно-ориентиран софтуер

- ▶ Инициатор в моделирането на обектно-ориентиран софтуер

