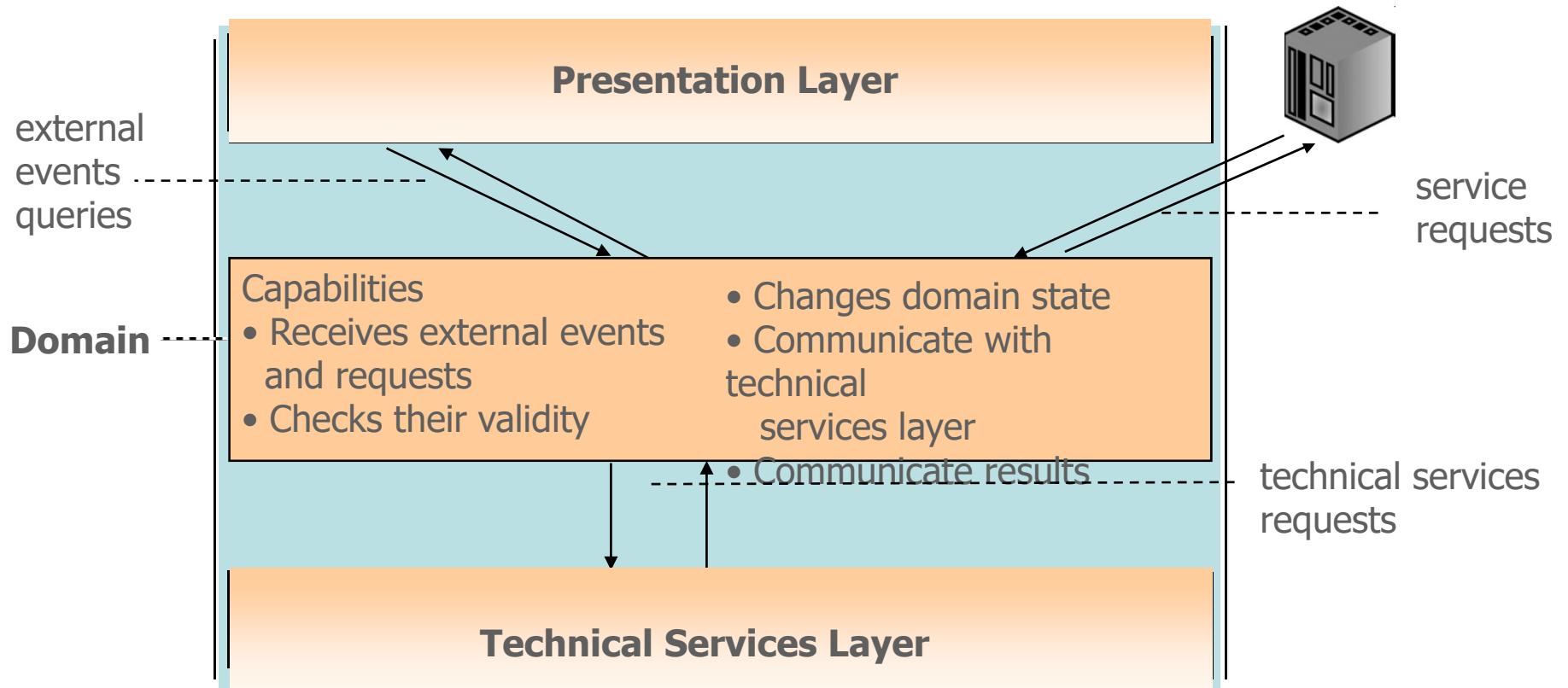


Domain Layer Design

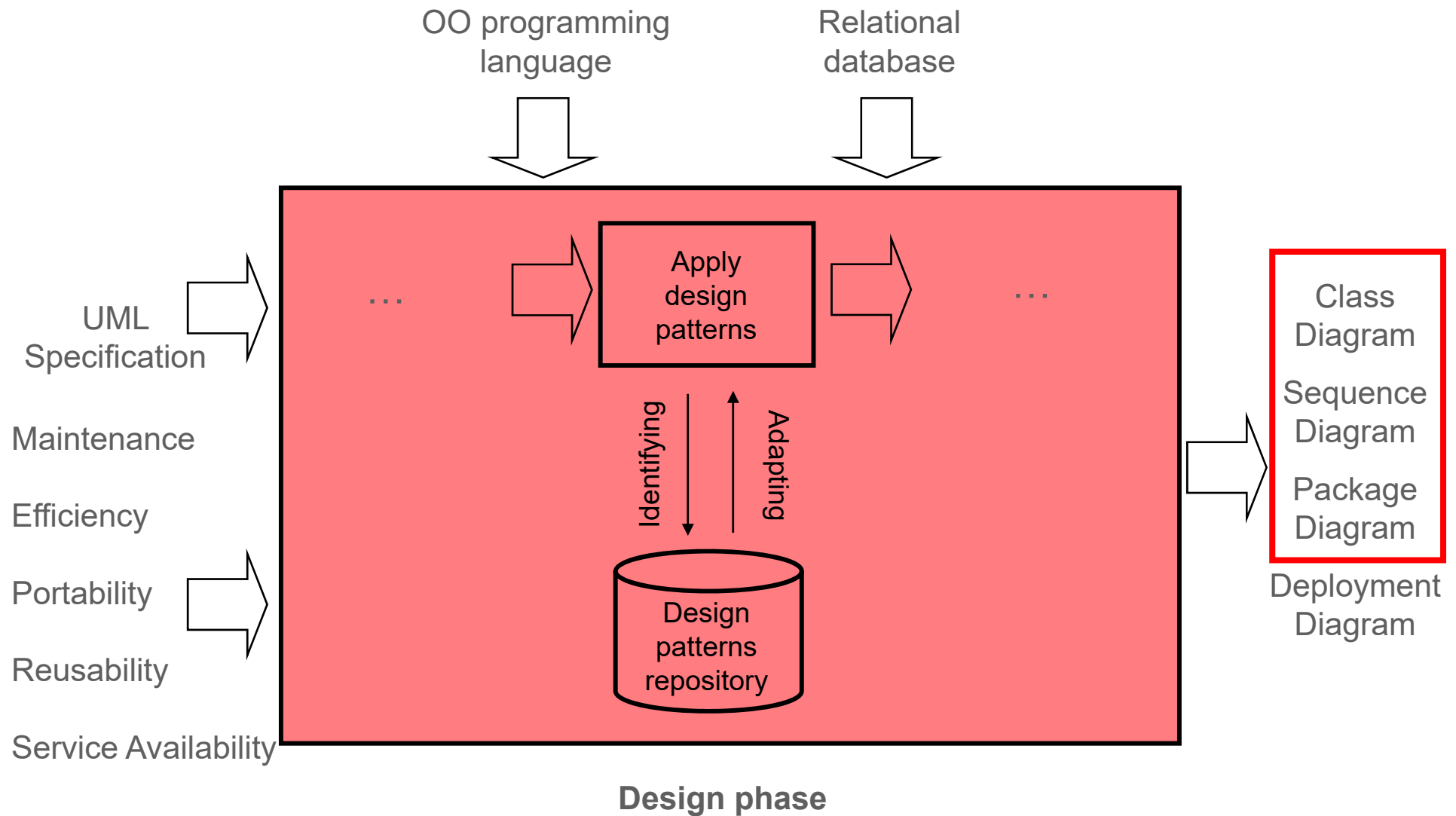
Domain Layer Design

- Pattern-based Design
- Patterns for Domain Layer
- Case Study
- References

Pattern-based Design



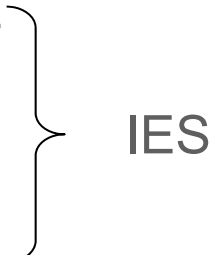
Pattern-based Design



Patterns for Domain Layer

- Patterns that determine the layers' structure. Proposed by Fowler (2003).
- Domain layer:
 - Great influence in the assignment of responsibilities to layers
 - Dominant patterns: *Domain Model*, *Transaction Script*
 - They determine the services and patterns that are offered by the data layer (*Data Mapper*, *Row Gateway*, *Active Record*)
 - Throughout this unit, we assume that *Domain Model* is applied

Patterns for Domain Layer

- General purpose-patterns that may be applied to the domain layer. Proposed by GoF (1995) and adapted by several authors to their own methods, e.g. Larman (2005).
 - Controller
 - Expert
 - State

IES

- Adapter
- Abstract Factory
- Singleton
- Strategy
- etc...

Case Study

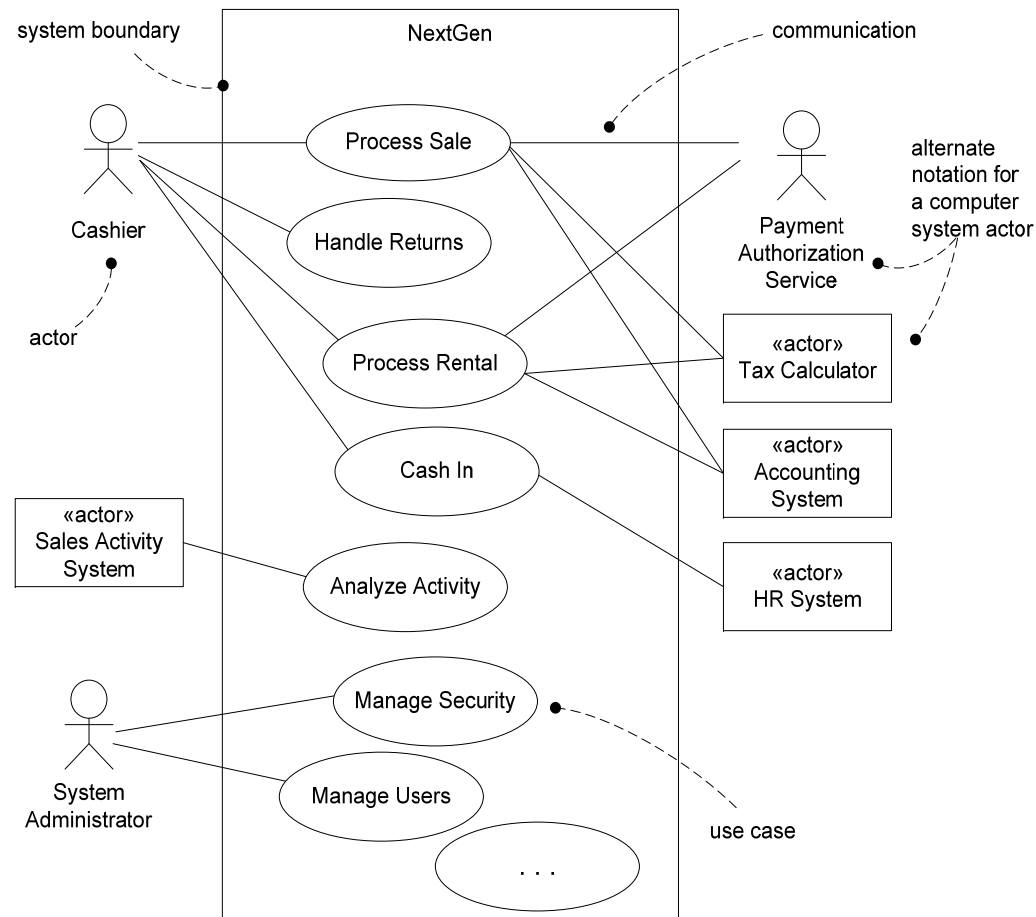


- NextGen is a point-of-sale system (POS) used to record sales and handle payments.
- It is typically used in a retail store. It includes hardware components as a computer and bar code scanner; and software to run the system.
- It interfaces to various service applications, such as a third-party tax calculator and inventory control. A POS system must be relatively fault-tolerant; that is even if remote services are temporarily unavailable (such as the inventory system), it must still be capable of capturing sales and handling at least cash payments.
- A POS system increasingly must support multiple and varied client-side terminals and interfaces.

Case Study



- Use Case Diagram

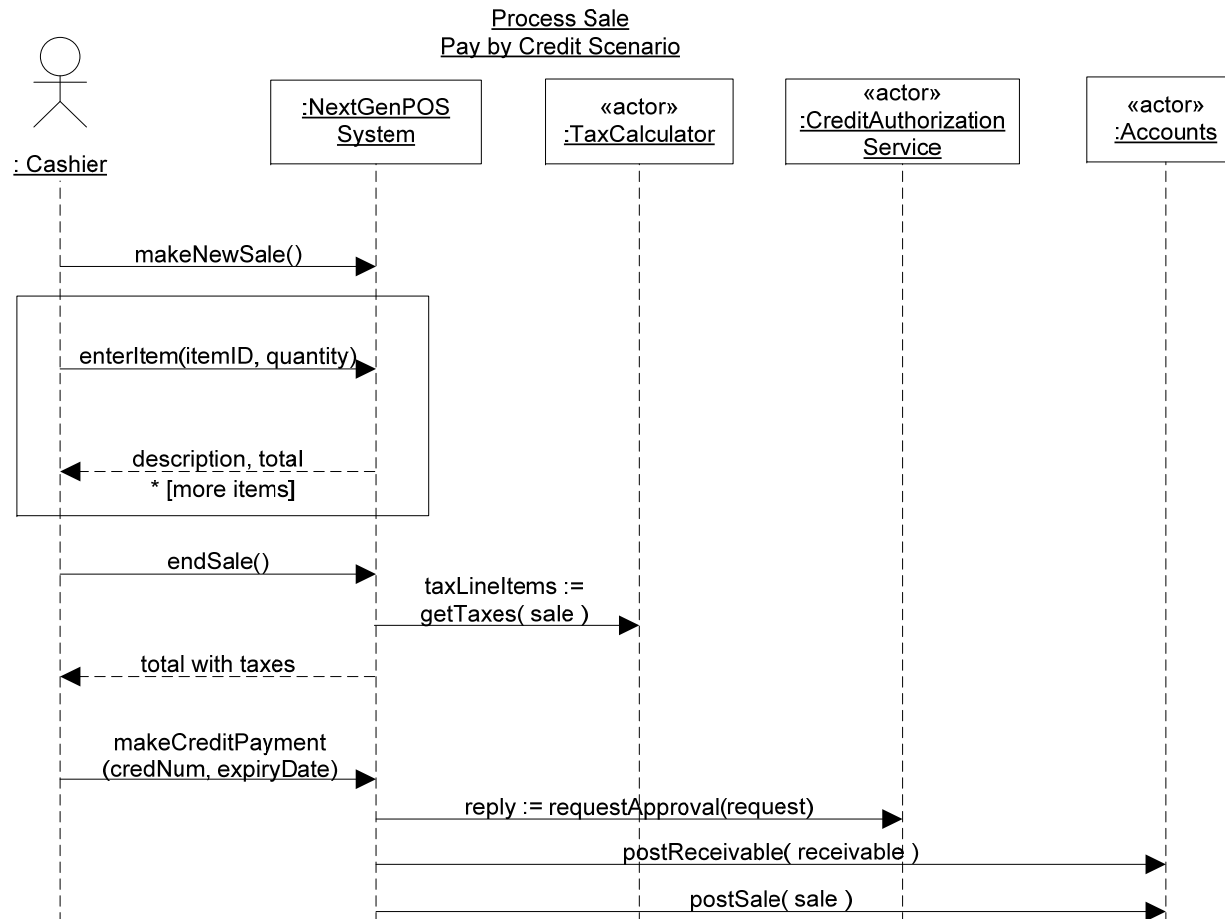


*Example extracted from Larman (2005)

Case Study



- System Sequence Diagram

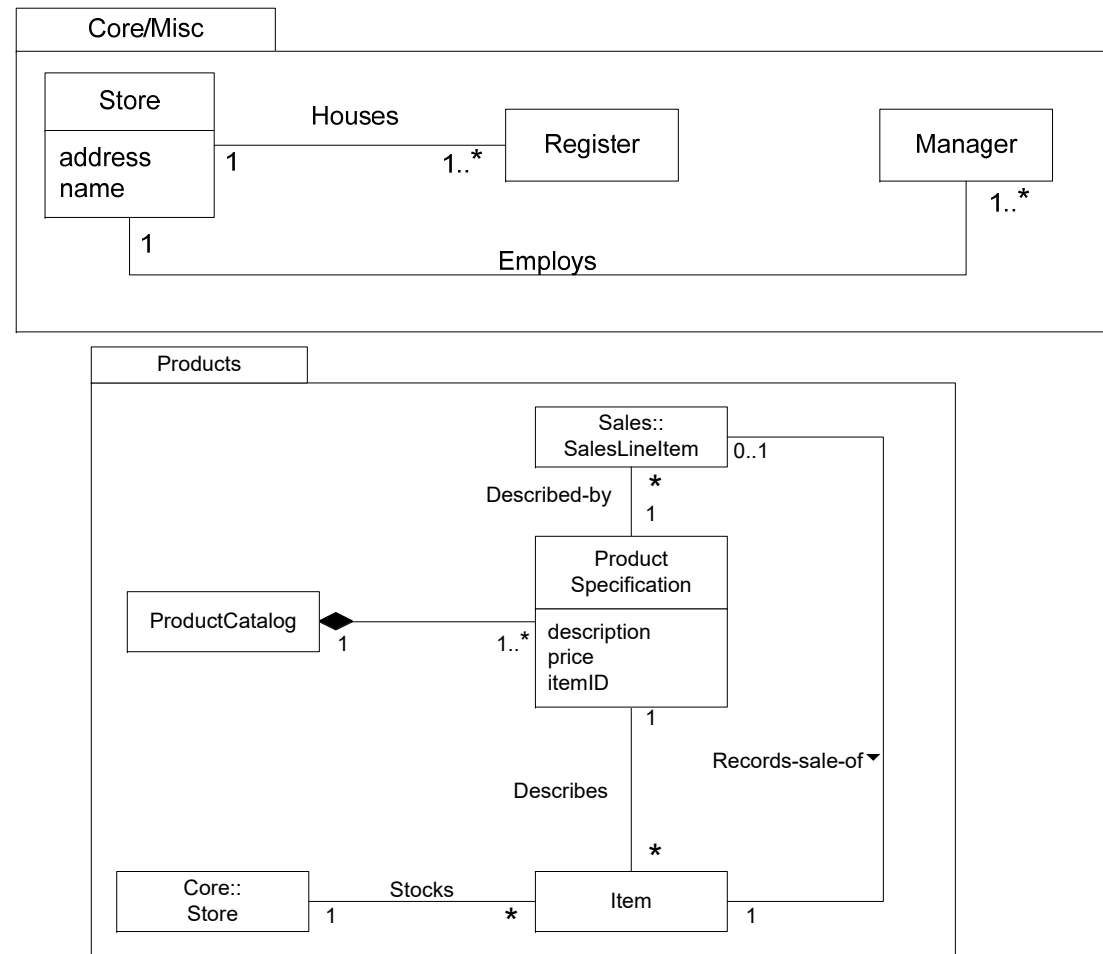


*Example extracted from Larman (2005)



Case Study

- Conceptual schema (partial)

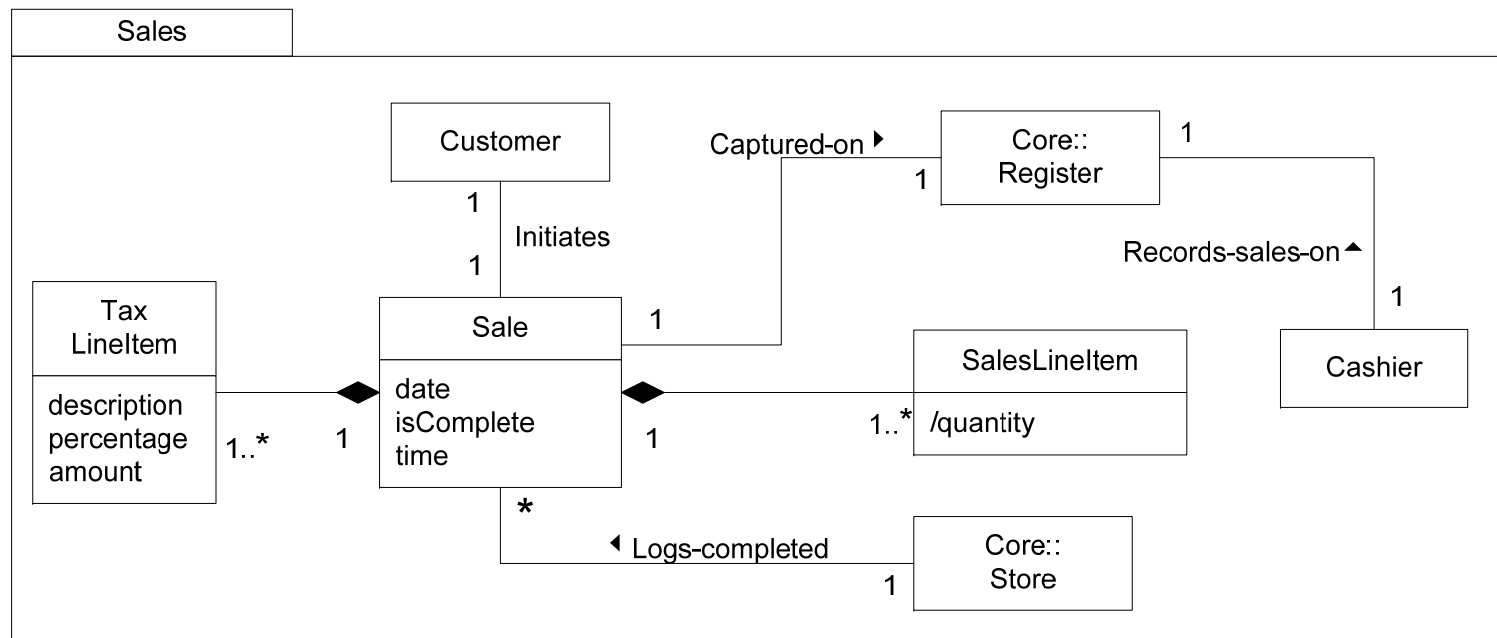


*Example extracted from Larman (2005)



Case Study

- Conceptual schema (partial)



*Example extracted from Larman (2005)

References

- *Design Patterns: Elements of Reusable Object-Oriented Software*
E. Gamma; R. Helm; R. Johnson; J. Vlissides
Addison-Wesley, 1995.
- *Patterns of Enterprise Application Architecture*
M. Fowler
Addison-Wesley, 2003
- *Applying UML and Patterns*
C. Larman
Prentice Hall, 2005 (Third edition), ch. 13 and 34
- *Agile Software Development: Principles, Patterns and Practices*
R.C. Martin
Prentice Hall, 2003