Calling Protocols and the Visitor

Zoran Horvat
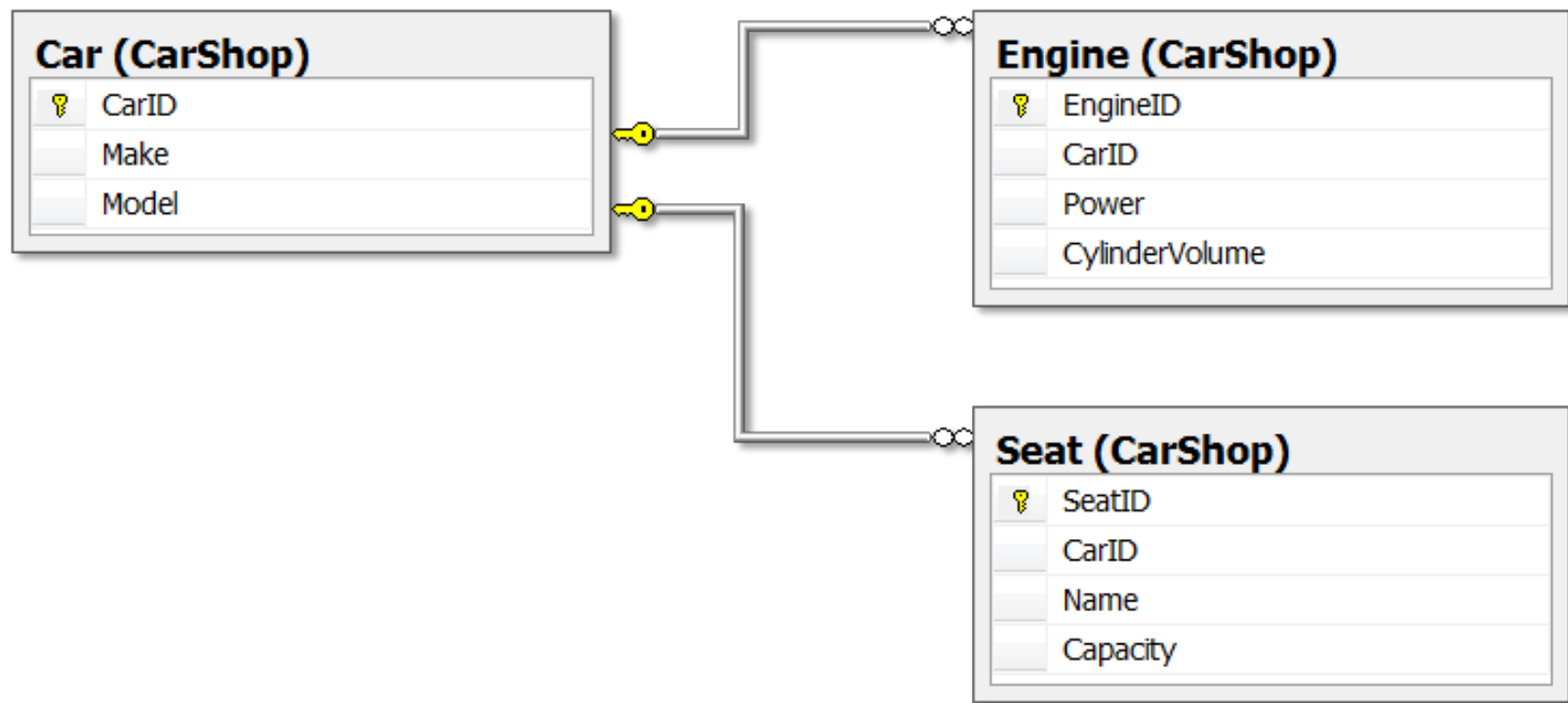
@zoranh75 | www.codinghelmet.com

The Encapsulation Issue

- Visitor requires access to internal data

- Encapsulation pushed to the limit
  - No raw data exposed to the outside world

- What happens when raw data are exposed through property getters?
  - Programmers tend to implement domain logic on the client side
  - This leads to problems
    - Code duplication
    - Maintenance problems
    - Bugs when multiple implementations are different

Command-Query Separation (CQS)

- There are two kinds of visitors
  - Ones are doing something
  - Others are producing an object
- There are two kinds of methods
  - Those that do something are commands
  - Those that return result are queries
- Command-Query Separation principle
  - Methods returning void are commands
    - Command methods change the system
  - Methods returning a result are queries
    - Query methods do not change the system

Car (CarShop)

🔑	CarID
	Make
	Model

Engine (CarShop)

🔑	EngineID
	CarID
	Power
	CylinderVolume

Seat (CarShop)

🔑	SeatID
	CarID
	Name
	Capacity

Summary

- Alternative implementations of the Visitor design pattern
  - It is not necessary to pass the Visitor object to the Accept methods
    - It is possible to send a factory method which produces the Visitor object
  - It is possible to provide two distinct Accept methods
    - One would return void and accept commands
    - Another would return an object and accept queries
  - Some complicated operations can completely be implemented in the Visitor
    - Domain class would then remain simple
- These alternatives are not part of the mainstream literature
- The next module: *Mixins*