

---

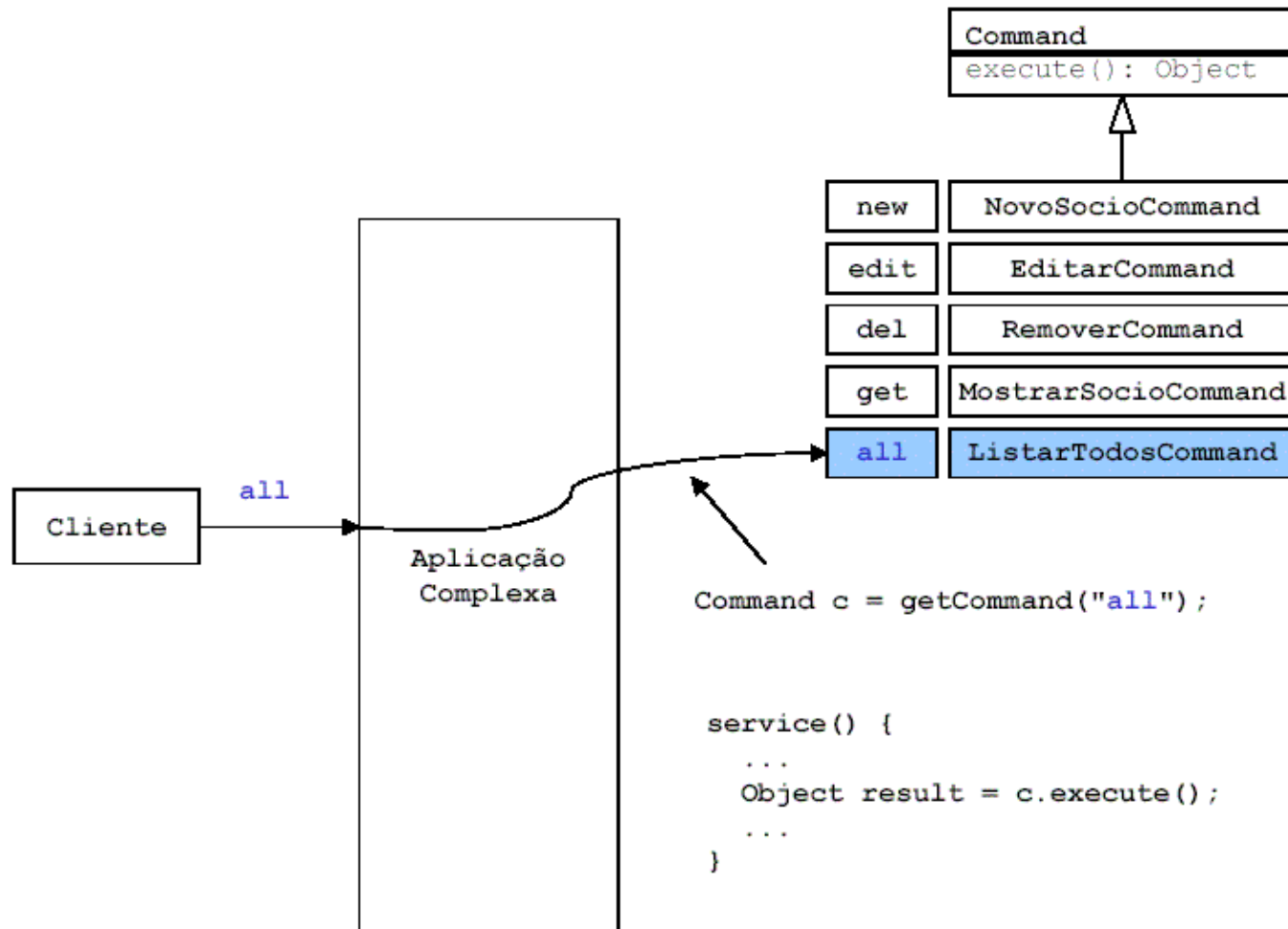
***Command***

# Command

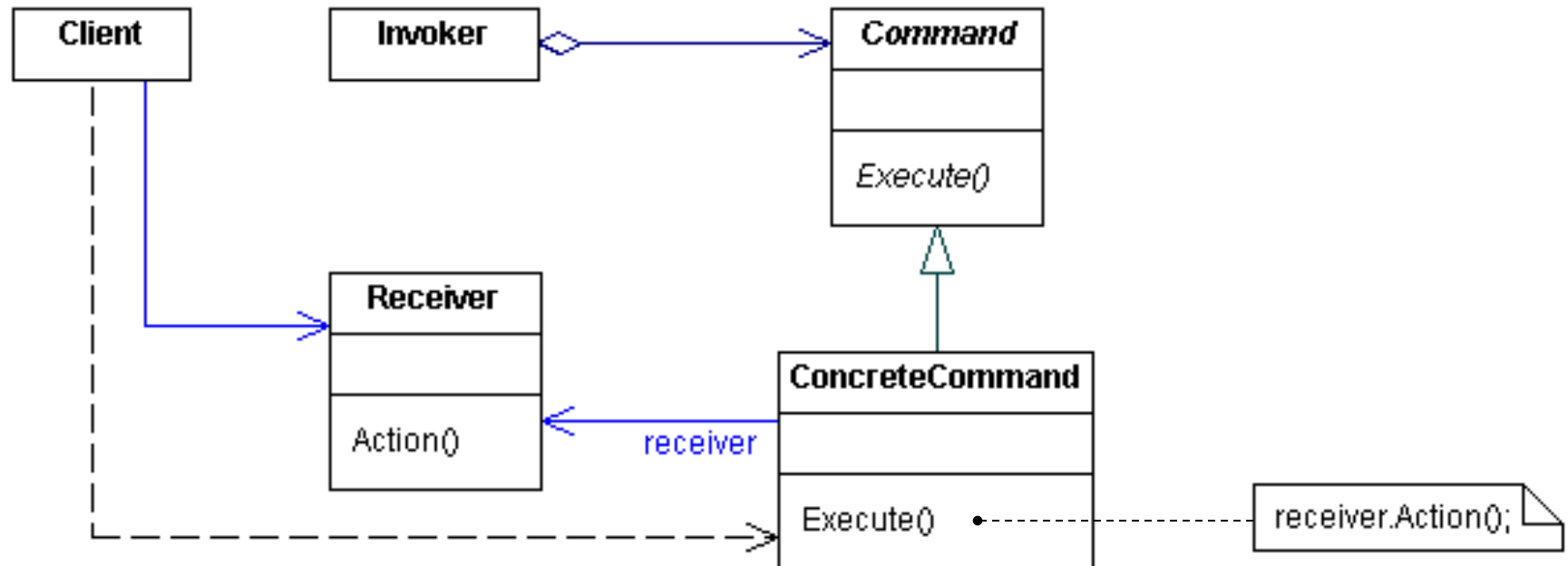
---

- Propósito
  - Encapsular uma requisição como um objeto, permitindo que clientes parametrizem diferentes requisições.
- Também conhecido como:
  - Action, Transaction

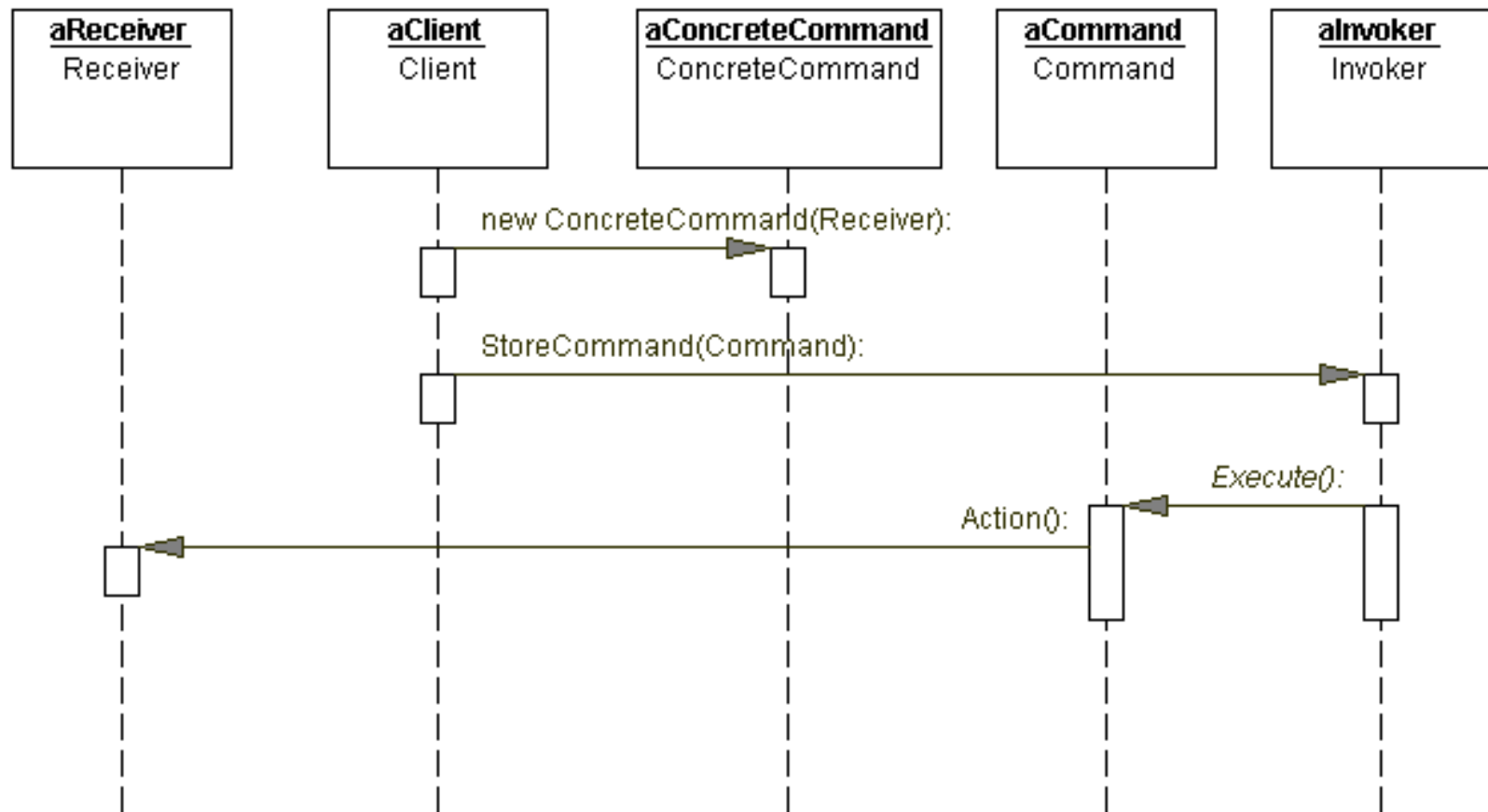
# Command - Motivação



# Command - Estrutura



# Command - Colaboração



# Command - Participantes

---

- Command
  - Define a interface para a execução de uma operação
- ConcreteCommand
  - Define uma vinculação entre o um objeto Receiver e uma ação
  - Implementa Execute através da invocação da(s) correspondente(s) operação(ões) no Receiver.
- Client
  - Cria um objeto ConcreteCommand e estabelece o seu receptor (Receiver).
- Invoker
  - Solicita ao Command a execução da solicitação
- Receiver
  - Sabe como executar as operações associadas a uma solicitação. Qualquer classe pode funcionar como um receiver.

# Command – Exemplo de código

```
public interface Command {  
    public Object execute(Object arg);  
}
```

```
public class Server {  
    private Database db = ...;  
    private HashMap cmds = new HashMap();  
  
    public Server() {  
        initCommands();  
    }  
  
    private void initCommands() {  
        cmds.put("new", new NewCommand(db));  
        cmds.put("del",  
                new DeleteCommand(db));  
        ...  
    }  
  
    public void service(String cmd,  
                        Object data) {  
        ...  
        Command c = (Command) cmds.get(cmd);  
        ...  
        Object result = c.execute(data);  
        ...  
    }  
}
```

```
public interface NewCommand implements Command {  
  
    public NewCommand(Database db) {  
        this.db = db;  
    }  
  
    public Object execute(Object arg) {  
        Data d = (Data) arg;  
        int id = d.getArg(0);  
        String nome = d.getArg(1);  
        db.insert(new Member(id, nome));  
    }  
}
```

```
public class DeleteCommand implements Command {  
  
    public DeleteCommand(Database db) {  
        this.db = db;  
    }  
  
    public Object execute(Object arg) {  
        Data d = (Data) arg;  
        int id = d.getArg(0);  
        db.delete(id);  
    }  
}
```