

# Engenharia de Software II

## Acoplamento Lógico

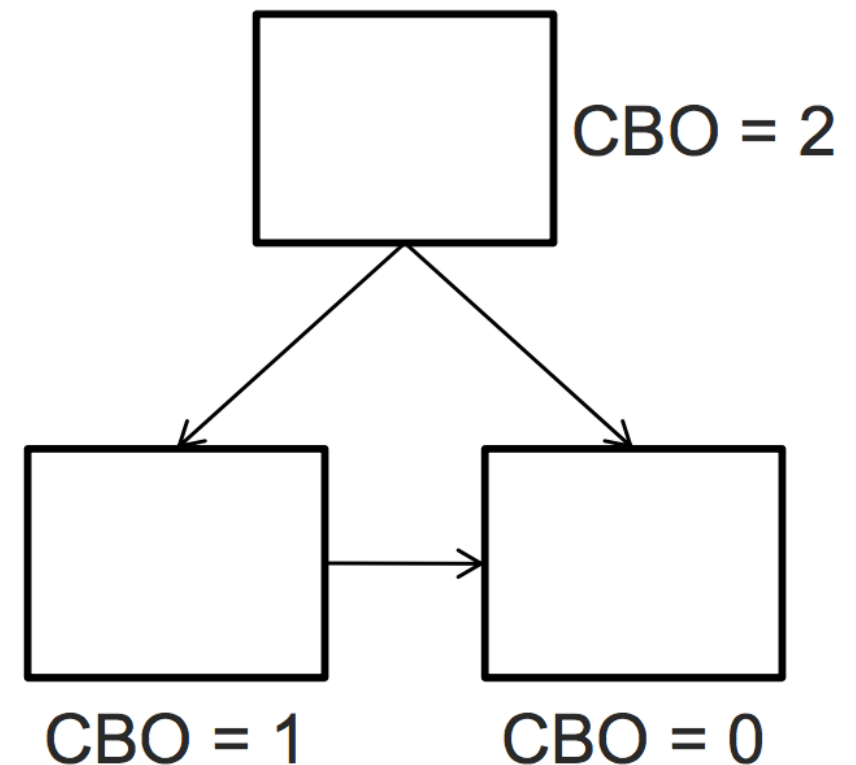
Prof. André Hora  
DCC/UFMG  
2019.1

# Acoplamento

- **Acoplamento:** dependências explícitas entre classes
  - Métricas de acoplamento (CK): CBO e RFC
- **Acoplamento lógico:** dependências implícitas entre classes
  - **Como detectar acoplamento lógico?**

# Acoplamento entre Objetos (CBO)

- Número de classes chamadas por uma classe
- Quanto mais acoplada uma classe, mais difícil de entender e manter



**Devemos buscar baixo acoplamento!**

# CBO?

## CBO: 5

- String
- List
- ArrayList
- Rental
- Movie

```
public class Customer {
    private String name;
    private List<Rental> rentals = new ArrayList<Rental>();
    public Customer(String name) {
        this.name = name;
    }
    public String getName() {
        return name;
    }
    public void addRental(Rental rental) {
        rentals.add(rental);
    }
    public String statement() {
        double totalAmount = 0;
        int f = 0;
        String r = "Rental record for " + getName() + "\n";
        for (Rental rental : rentals) {
            double amount = 0;
            switch (rental.getMovie().getPriceCode()) {
                case Movie.REGULAR:
                    amount += 2;
                    if (rental.getDaysRented() > 2)
                        amount += (rental.getDaysRented() - 2) * 1.5;
                    break;
                case Movie.NEW_RELEASE:
                    amount += rental.getDaysRented() * 3;
                    break;
                case Movie.CHILDREN:
                    amount += 1.5;
                    if (rental.getDaysRented() > 3)
                        amount += (rental.getDaysRented() - 3) * 1.5;
                    break;
            }
            f++;
            if (rental.getMovie().getPriceCode() == Movie.NEW_RELEASE)
                f++;
            r += "\t" + rental.getMovie().getTitle() + "\t" + String.valueOf(amount) + "\n";
            totalAmount += amount;
        }
        r += "Amount owed is " + String.valueOf(totalAmount) + "\n";
        r += "You earned " + String.valueOf(f) + " frequent renter points";
        return r;
    }
}
```

# Acoplamiento Lógico

```
commit cfa4ad4132ec34d8a18e6beee1aaa136a2e4f3ba ← Second commit
Author: Adam Tornhill <adam@adamtornhill.com>
Date:   Wed Nov 13 07:20:43 2013 +0100
```

Support multiple grades of fuel (small fix, right).

32	4	src/detectors/FuelGradeDetector.cs
22	7	src/engine/FuelInjector.cs
2	1	src/engine/Engine.cs
2	1	src/statistics/Diagnostics.cs
11	2	test/engine/FuelInjectorTest.cs

Modules that keep  
changing together

```
commit 314d56b5d0018dde11aeb8a34521fd8bb1a70aa0 ← First commit
Author: Adam Tornhill <adam@adamtornhill.com>
Date:   Tue Nov 12 17:04:26 2013 +0100
```

More efficient fuel injection by higher pressure.

18	2	src/engine/FuelInjector.cs
2	1	src/engine/Engine.cs
2	1	src/statistics/Diagnostics.cs
10	3	test/engine/FuelInjectorTest.cs

# Acoplamiento Lógico

```
commit cfa4ad4132ec34d8a18e6beee1aaa136a2e4f3ba ← Second commit
Author: Adam Tornhill <adam@adamtornhill.com>
Date:   Wed Nov 13 07:20:43 2013 +0100
```

Support multiple grades of fuel (small fix, right).

32	4	src/detectors/FuelGradeDetector.cs
22	7	src/engine/FuelInjector.cs
2	1	src/engine/Engine.cs
2	1	src/statistics/Diagnostics.cs
11	2	test/engine/FuelInjectorTest.cs

Modules that keep  
changing together

```
commit 314d56b5d0018dde11aeb8a34521fd8bb1a70aa0 ← First commit
Author: Adam Tornhill <adam@adamtornhill.com>
Date:   Tue Nov 12 17:04:26 2013 +0100
```

More efficient fuel injection by higher pressure.

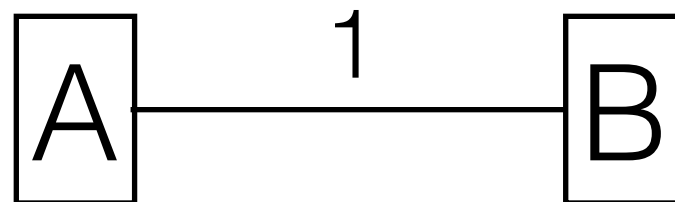
18	2	src/engine/FuelInjector.cs
2	1	src/engine/Engine.cs
2	1	src/statistics/Diagnostics.cs
10	3	test/engine/FuelInjectorTest.cs

# Acoplamento Lógico

- Grafo ponderado  $\{V, E\}$  onde  $V$  é um conjunto de classes e  $E$  um conjunto de arestas
- Se existe uma aresta entre dois vértices  $V_i$  e  $V_j$ , então existe pelo menos um commit onde  $V_i$  e  $V_j$  foram alterados
- O peso da aresta representa a quantidade total de alterações entre as classes

Commit 1: A, B

# Exemplo

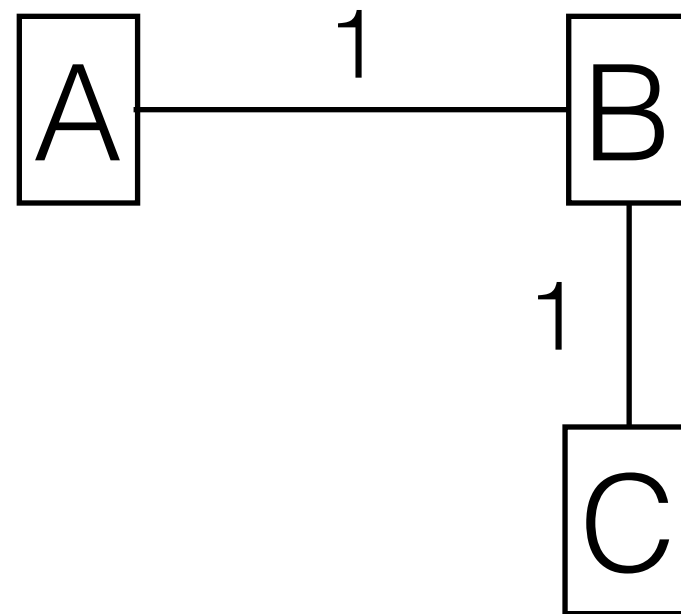




Commit 1: A, B

Commit 2: B, C

# Exemplo

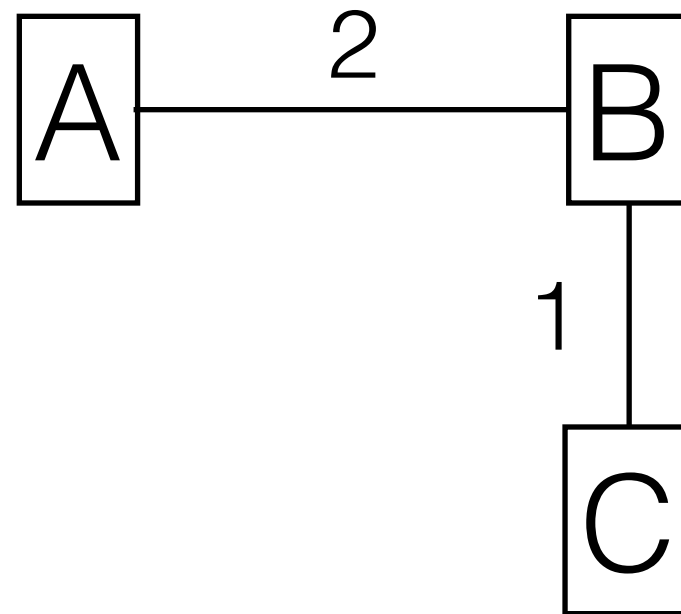


Commit 1: A, B

Commit 2: B, C

Commit 3: A, B

# Exemplo



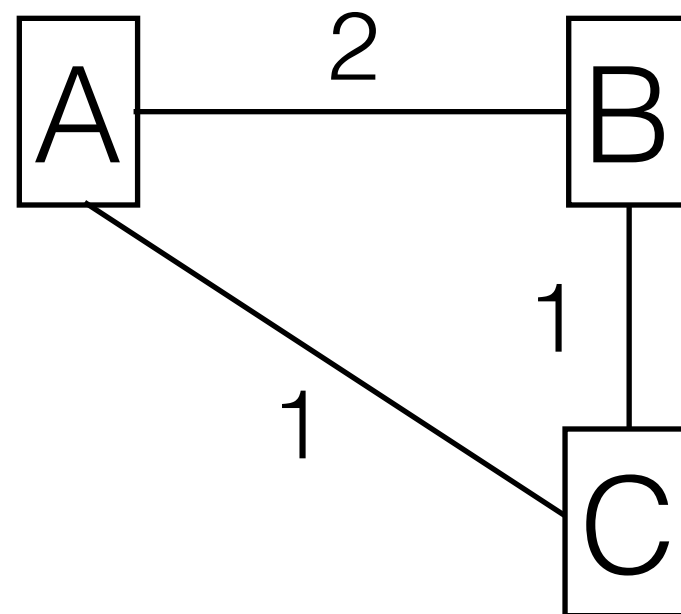
Commit 1: A, B

Commit 2: B, C

Commit 3: A, B

Commit 4: A, C

# Exemplo



# Exemplo

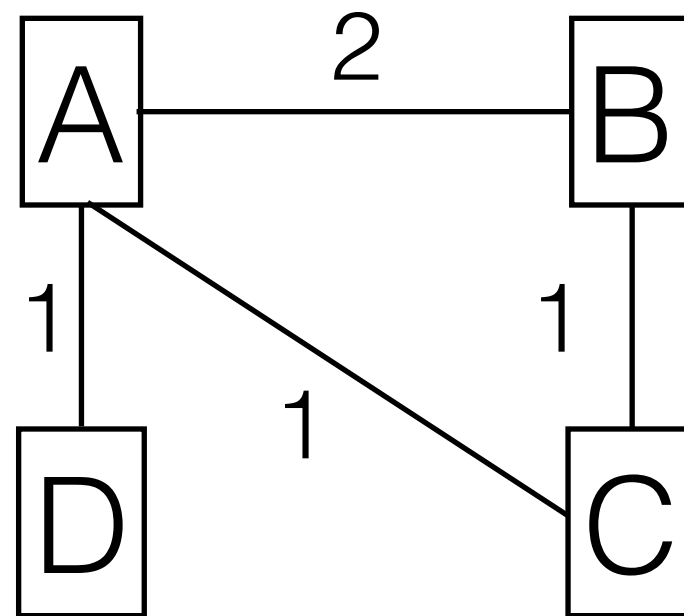
Commit 1: A, B

Commit 2: B, C

Commit 3: A, B

Commit 4: A, C

Commit 5: A, D



# Exercício

Considerando o log de alterações de código abaixo, apresente:

- a)** Os sete grafos intermediários de acoplamento temporal (do commit 1 ao 7).
- b)** O conjunto de classes mais acopladas.

1. 50d0b4df46  
OnClassCondition.java  
InfoEndpoint.java

2. 19275fd9d6  
SessionProperties.java  
OnClassCondition.java  
InfoEndpoint.java

3. 28cf63354a  
SingleKeyEntry.java

4. 078bd890a6  
SessionProperties.java  
InfoEndpoint.java

5. 51013421f2  
SingleKeyEntry2.java

6. 161c0b9fad  
RestTemplate.java  
TestRestTemplate.java

7. a44a55e948  
RestTemplate.java  
SessionProperties.java  
InfoEndpoint.java  
OnClassCondition.java

# Considerando o log de alterações de código abaixo, apresente:

**a)** Os sete grafos intermediários de acoplamento temporal (do commit 1 ao 7).

1. 50d0b4df46  
OnClassCondition.java  
InfoEndpoint.java

2. 19275fd9d6  
SessionProperties.java  
OnClassCondition.java  
InfoEndpoint.java

3. 28cf63354a  
SingleKeyEntry.java

4. 078bd890a6  
SessionProperties.java  
InfoEndpoint.java

5. 51013421f2  
SingleKeyEntry2.java

6. 161c0b9fad  
RestTemplate.java  
TestRestTemplate.java

7. a44a55e948  
RestTemplate.java  
SessionProperties.java  
InfoEndpoint.java  
OnClassCondition.java

# Considerando o log de alterações de código abaixo, apresente:

**a)** Os sete grafos intermediários de acoplamento temporal (do commit 1 ao 7).

1

1. 50d0b4df46  
OnClassCondition.java  
InfoEndpoint.java

2. 19275fd9d6  
SessionProperties.java  
OnClassCondition.java  
InfoEndpoint.java

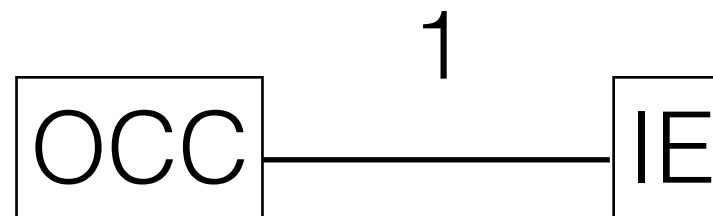
3. 28cf63354a  
SingleKeyEntry.java

4. 078bd890a6  
SessionProperties.java  
InfoEndpoint.java

5. 51013421f2  
SingleKeyEntry2.java

6. 161c0b9fad  
RestTemplate.java  
TestRestTemplate.java

7. a44a55e948  
RestTemplate.java  
SessionProperties.java  
InfoEndpoint.java  
OnClassCondition.java



# Considerando o log de alterações de código abaixo, apresente:

**a)** Os sete grafos intermediários de acoplamento temporal (do commit 1 ao 7).

2

1. 50d0b4df46  
**OnClassCondition.java**  
**InfoEndpoint.java**

2. 19275fd9d6  
**SessionProperties.java**  
**OnClassCondition.java**  
**InfoEndpoint.java**

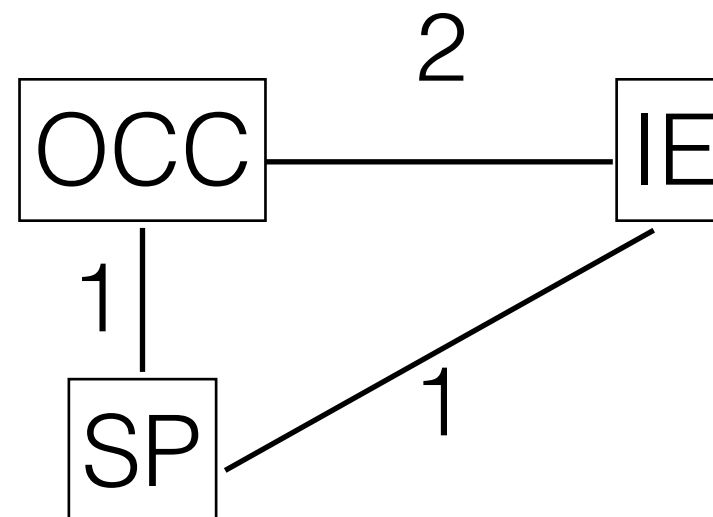
3. 28cf63354a  
SingleKeyEntry.java

4. 078bd890a6  
SessionProperties.java  
InfoEndpoint.java

5. 51013421f2  
SingleKeyEntry2.java

6. 161c0b9fad  
RestTemplate.java  
TestRestTemplate.java

7. a44a55e948  
RestTemplate.java  
SessionProperties.java  
InfoEndpoint.java  
OnClassCondition.java





# Considerando o log de alterações de código abaixo, apresente:

**a)** Os sete grafos intermediários de acoplamento temporal (do commit 1 ao 7).

3

1. 50d0b4df46  
**OnClassCondition.java**  
**InfoEndpoint.java**

2. 19275fd9d6  
**SessionProperties.java**  
**OnClassCondition.java**  
**InfoEndpoint.java**

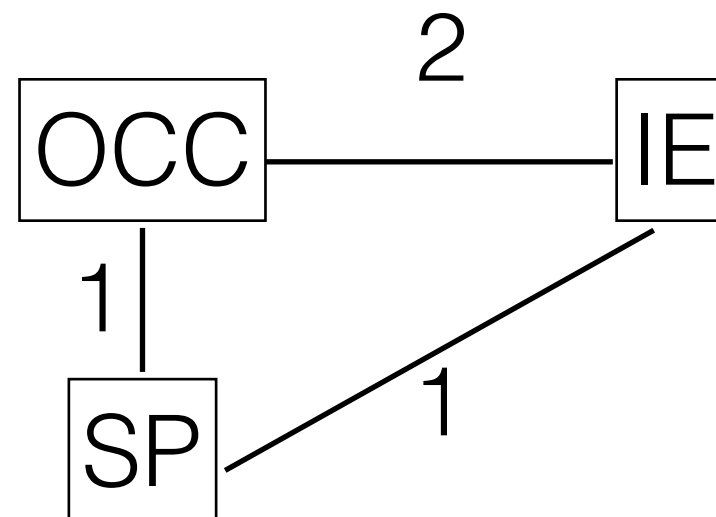
3. 28cf63354a  
**SingleKeyEntry.java**

4. 078bd890a6  
**SessionProperties.java**  
**InfoEndpoint.java**

5. 51013421f2  
**SingleKeyEntry2.java**

6. 161c0b9fad  
**RestTemplate.java**  
**TestRestTemplate.java**

7. a44a55e948  
**RestTemplate.java**  
**SessionProperties.java**  
**InfoEndpoint.java**  
**OnClassCondition.java**



SKE

# Considerando o log de alterações de código abaixo, apresente:

**a)** Os sete grafos intermediários de acoplamento temporal (do commit 1 ao 7).

4

1. 50d0b4df46  
OnClassCondition.java  
InfoEndpoint.java

2. 19275fd9d6  
SessionProperties.java  
OnClassCondition.java  
InfoEndpoint.java

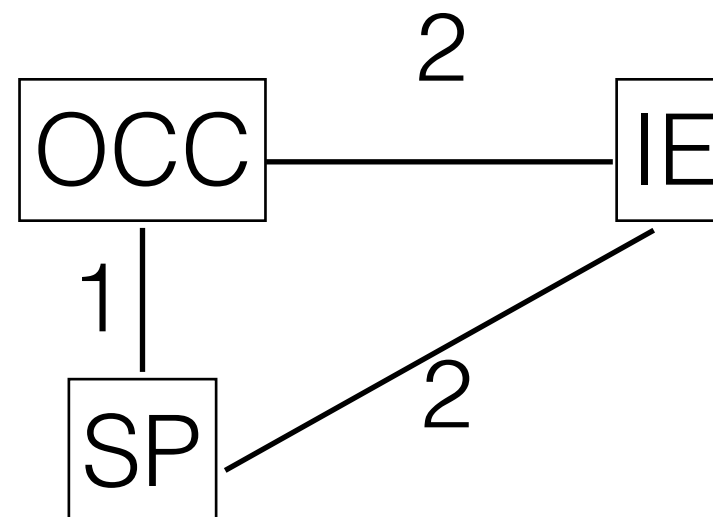
3. 28cf63354a  
SingleKeyEntry.java

4. 078bd890a6  
SessionProperties.java  
InfoEndpoint.java

5. 51013421f2  
SingleKeyEntry2.java

6. 161c0b9fad  
RestTemplate.java  
TestRestTemplate.java

7. a44a55e948  
RestTemplate.java  
SessionProperties.java  
InfoEndpoint.java  
OnClassCondition.java



SKE

# Considerando o log de alterações de código abaixo, apresente:

**a)** Os sete grafos intermediários de acoplamento temporal (do commit 1 ao 7).

5

1. 50d0b4df46  
OnClassCondition.java  
InfoEndpoint.java

2. 19275fd9d6  
SessionProperties.java  
OnClassCondition.java  
InfoEndpoint.java

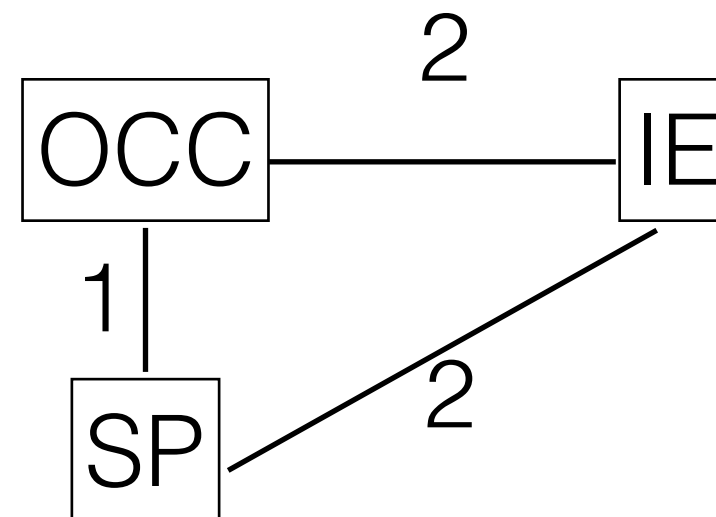
3. 28cf63354a  
SingleKeyEntry.java

4. 078bd890a6  
SessionProperties.java  
InfoEndpoint.java

5. 51013421f2  
SingleKeyEntry2.java

6. 161c0b9fad  
RestTemplate.java  
TestRestTemplate.java

7. a44a55e948  
RestTemplate.java  
SessionProperties.java  
InfoEndpoint.java  
OnClassCondition.java



# Considerando o log de alterações de código abaixo, apresente:

a) Os sete grafos intermediários de acoplamento temporal (do commit 1 ao 7).

6

1. 50d0b4df46  
OnClassCondition.java  
InfoEndpoint.java

2. 19275fd9d6  
SessionProperties.java  
OnClassCondition.java  
InfoEndpoint.java

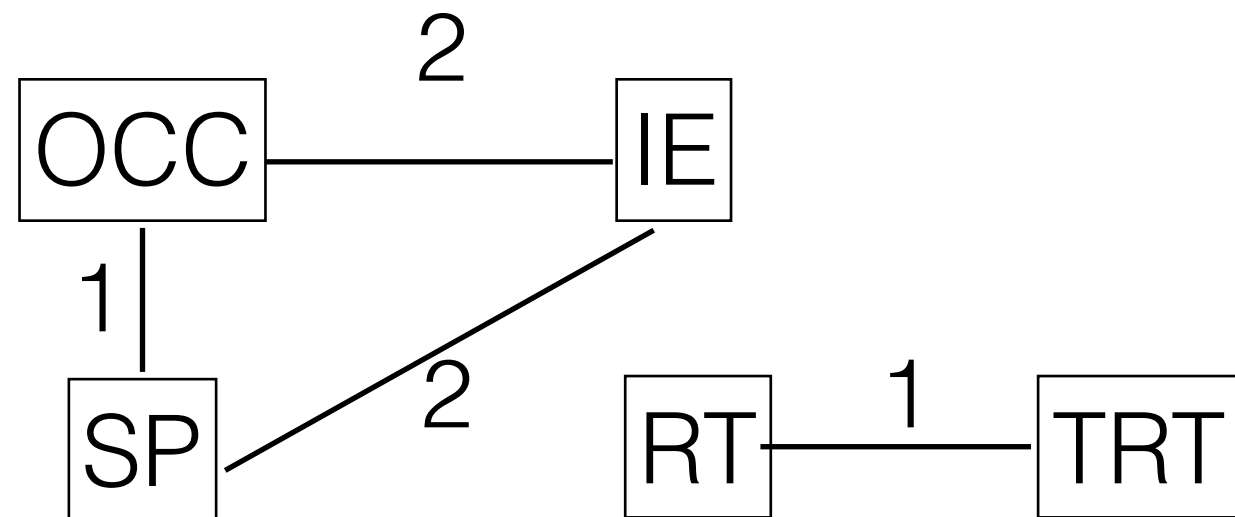
3. 28cf63354a  
SingleKeyEntry.java

4. 078bd890a6  
SessionProperties.java  
InfoEndpoint.java

5. 51013421f2  
SingleKeyEntry2.java

6. 161c0b9fad  
RestTemplate.java  
TestRestTemplate.java

7. a44a55e948  
RestTemplate.java  
SessionProperties.java  
InfoEndpoint.java  
OnClassCondition.java



# Considerando o log de alterações de código abaixo, apresente:

a) Os sete grafos intermediários de acoplamento temporal (do commit 1 ao 7).

7

1. 50d0b4df46  
OnClassCondition.java  
InfoEndpoint.java

2. 19275fd9d6  
SessionProperties.java  
OnClassCondition.java  
InfoEndpoint.java

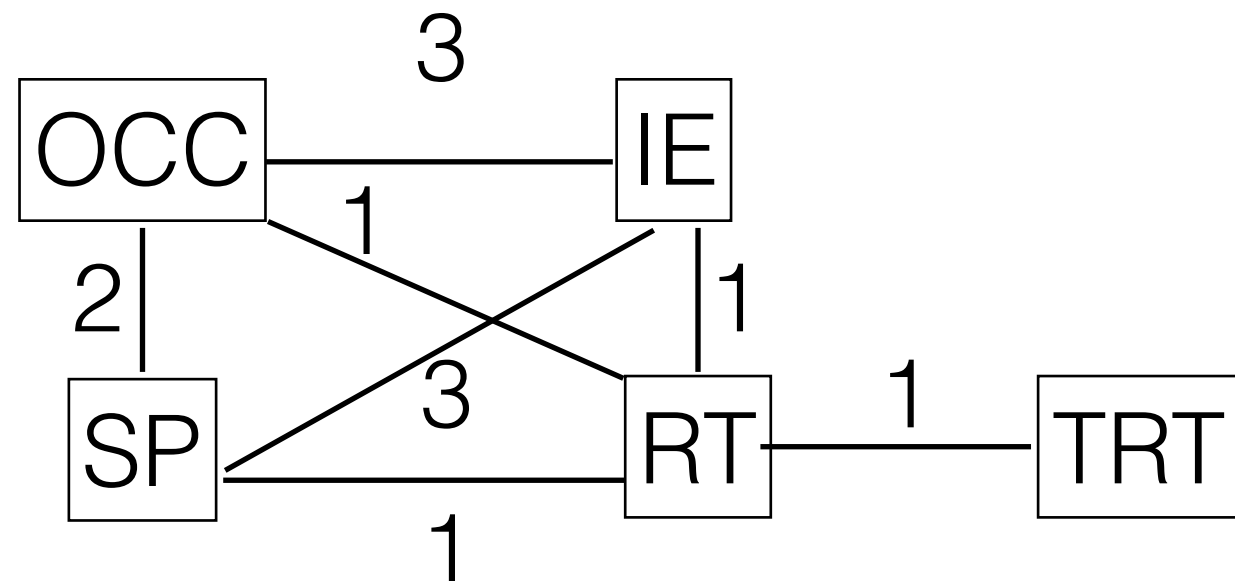
3. 28cf63354a  
SingleKeyEntry.java

4. 078bd890a6  
SessionProperties.java  
InfoEndpoint.java

5. 51013421f2  
SingleKeyEntry2.java

6. 161c0b9fad  
RestTemplate.java  
TestRestTemplate.java

7. a44a55e948  
RestTemplate.java  
SessionProperties.java  
InfoEndpoint.java  
OnClassCondition.java



# Considerando o log de alterações de código abaixo, apresente:

**b)** O conjunto de classes mais acopladas.

1. 50d0b4df46  
OnClassCondition.java  
InfoEndpoint.java

2. 19275fd9d6  
SessionProperties.java  
OnClassCondition.java  
InfoEndpoint.java

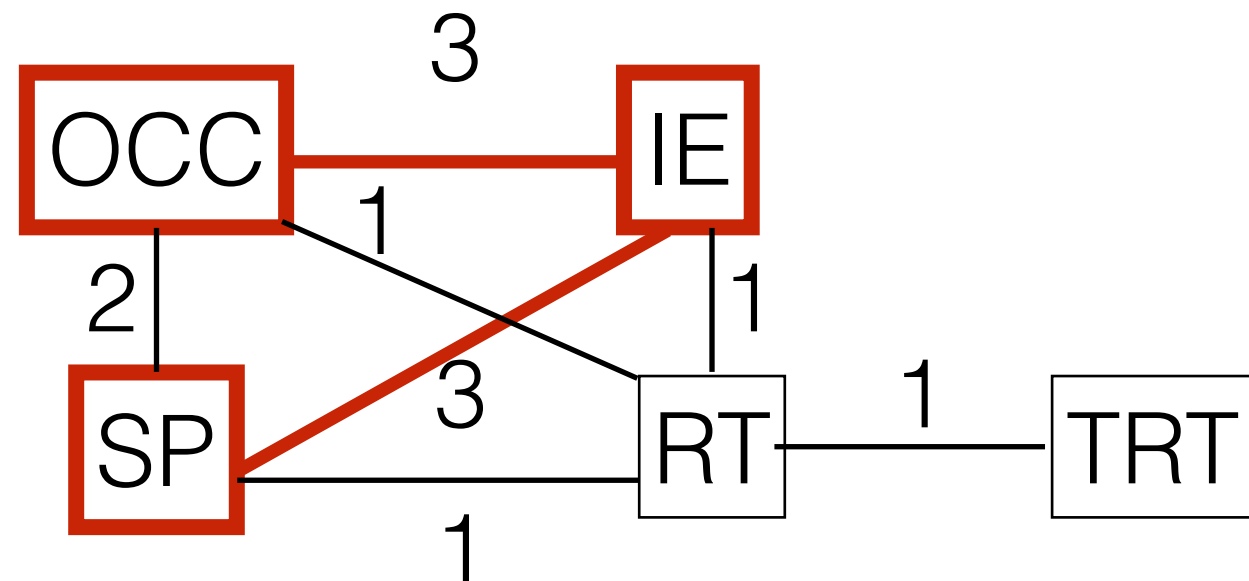
3. 28cf63354a  
SingleKeyEntry.java

4. 078bd890a6  
SessionProperties.java  
InfoEndpoint.java

5. 51013421f2  
SingleKeyEntry2.java

6. 161c0b9fad  
RestTemplate.java  
TestRestTemplate.java

7. a44a55e948  
RestTemplate.java  
SessionProperties.java  
InfoEndpoint.java  
OnClassCondition.java



SKE

SKE2