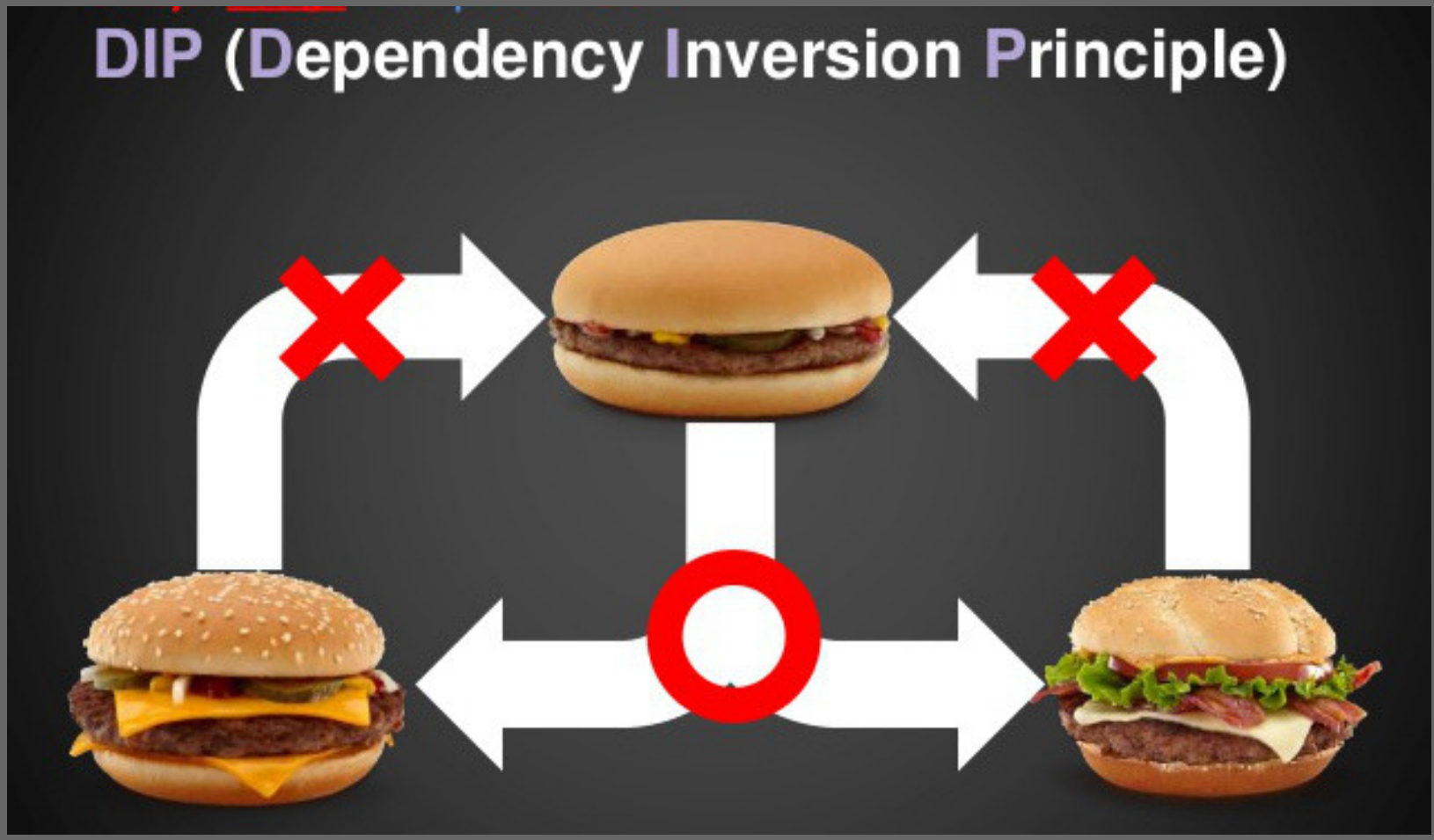
DEPENDENCY INVERSION

IN PERSPECTIVE

The Basics



The good, the Bad and the Ugly!!

```
/ Dependency Inversion Principle - Bad example

class CheeseBurger {
  public void addDressing() {
  // ....cheese and sauce }
}

class HamBurger {
  public void addDressing() {
  //.... Ham sauce}
}

public class Customer {
  CheeseBurger cheeseBurger;
  HamBurger hamBurger;

  public Customer(CheeseBurger cheeseBurger,
  HamBurger hamBurger);
  this.cheeseBurger = cheeseBurger;
  this.hamBurger = hamBurger;

  public void setBurger(CheeseBurger
    cheeseBurger) {
    this.cheeseBurger = cheeseBurger;
  }

  public void orderCheese() {
   cheeseBurger.addDressing();}

  public void setBurger(HamBurger
    hamBurger) {
    this.hamBurger = hamBurger; }

  public void orderHam() {
    hamBurger.addDressing();}
  }
```

The good, the Bad and the Ugly!!

```
// Dependency Inversion Principle - Good example
interface Burger {
 public void addDressing();
}

class HamBurger implements Burger{
 public void addDressing() {
   // ....sauce  }
}

class CheeseBurger  implements Burger{
 public void addDressing() {
   //.... cheese and sauce  }
}

class Customer {

 Burger burger;
```

```
public Customer(Burger burger) {
  this.burger = burger;
}

public void setBurger(Burger
burger) {
   this.burger = burger;
}

public void order() {
  burger.addDressing();  }
}

HamBurger ham = new
HamBurger();
CheeseBurger cheese = new
CheeseBurger();

customer = new Customer(ham);
customer.setBurger(cheese);
```


McKenn
a
Jones
& Young Production