

# Object-Z

Christofer Chávez Carazas    Juan León Camilo

Universidad Nacional de San Agustín

25 de noviembre de 2016

- Object-Z es una extensión orientada a objetos del lenguaje de especificación Z.
- Object-Z añade a Z formas de expresar conceptos del paradigma orientado a objetos, sobre todo clases, polimorfismo y herencia.

# Ejemplo de las Tarjetas de crédito

- El sistema está previsto de una colección de cuentas de tarjetas de crédito.
- Cada cuenta tiene dos números, el balance actual y el crédito límite.
- El dinero puede ser retirado o depositado.



Figura: Class

† (limit, balance, Init, withdraw, deposit, withdrawAvail)

## Figura: Visibility list

- Son las partes y componentes que son visibles en el entorno de un objeto Tarjeta de Crédito.
- Si la visibility list es omitida, se deducirá que todos los componentes de la clase son visibles.

# Definición de constantes

limit: N
limit $\in \{1000, 2000, 5000\}$

Figura: Constantes

- Se grafica con una caja abierta.
- La parte de arriba es una típica declaración en Z de una lista.
- Y la parte de abajo los valores de la lista.

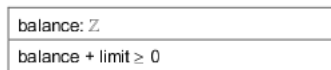


Figura: State Schema

- Se grafica con una caja cerrada sin nombre.
- Se utiliza el mismo patrón que en la definición de las constantes.
- Aquí se definen las variables.

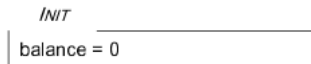


Figura: Initial Schema

- Se grafica con una caja cerrada nombrada con la palabra *INIT*.
- Se inicializan las variables.
- El predicado inicial debe ser verdadero para todas las variables.



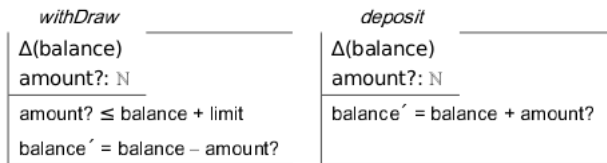


Figura: Operation Schema

- Los esquemas se definen de la misma forma que el lenguaje Z.

# TwoCards

## TwoCards

$\vdash (totalbal, INIT, withdraw_1, transfer, withdrawEither, replaceCard_1, \dots)$

$c_1, c_2 : CreditCard$

$\Delta$

$totalbal : \mathbb{Z}$

$c_1 \neq c_2$

$totalbal = c_1.balance + c_2.balance$

*INIT*

$c_1.INIT$

$c_2.INIT$

$withdraw_1 \triangleq c_1.withdraw$

$transfer \triangleq c_1.withdraw \wedge c_2.deposit$

$withdrawEither \triangleq c_1.withdraw \parallel c_2.withdraw$

$transferAvail \triangleq c_1.withdrawAvail \parallel c_2.deposit$

*replaceCard<sub>1</sub>*

$\Delta(c_1)$

$card? : CreditCard$

$card? \notin \{c_1, c_2\}$

$card?.limit = c_1.limit$

$card?.balance = c_1.balance$

$c_1' = card?$

[plus other operations not specified here]

Figura: TwoCards

# Variables secundarias

$C_1, C_2 : \text{CreditCard}$
$\Delta$
$\text{totalbal} : \mathbb{Z}$
$C_1 \neq C_2$
$\text{totalbal} = c_1.\text{balance} + c_2.\text{balance}$

Figura: Secondary Variables

Las variables secundarias son declaradas en terminos de las variables primarias.

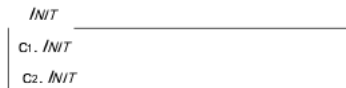


Figura: Inital Schema

$\text{withdraw}_1 \triangleq c_1.\text{withdraw}$

$\text{transfer} \triangleq c_1.\text{withdraw} \wedge c_2.\text{deposit}$

$\text{withdrawEither} \triangleq c_1.\text{withdraw} \sqcup c_2.\text{withdraw}$

$\text{transferAvail} \triangleq c_1.\text{withdrawAvail} \parallel c_2.\text{deposit}$

Figura: Operation expressions

- **Expresion de Conjunción:** Símbolo:  $\wedge$
- **Expresion de Elección:** Símbolo:  $\square$
- **Composición Paralela:** Símbolo:  $\parallel$
- **Composición Secuencial:** Símbolo:  $;$

## CreditCards

$\vdash (\text{commonlimit}, \text{INIT}, \text{add}, \text{delete}, \text{withdraw}, \text{deposit}, \text{withdrawAvail}, \text{transferAvail})$

$\text{commonlimit} : \mathbb{N}$

$\text{commonlimit} \in \{1000, 2000, 5000\}$

$\text{cards} : \mathbb{P} \text{CreditCard}$

$\forall c:\text{cards} \bullet c.\text{limit} = \text{commonlimit}$

*INIT*

$\text{cards} = \emptyset$

*add*

$\Delta(\text{cards})$

$\text{card}?: \text{CreditCard}$

$\text{card} \notin \text{cards}$

$\text{card}?.\text{limit} = \text{commonlimit}$

$\text{card}?.\text{INIT}$

$\text{cards}' = \text{cards} \cup \{\text{card}?\}$

*delete*

$\Delta(\text{cards})$

$\text{card}?: \text{CreditCard}$

$\text{card} \in \text{cards}$

$\text{cards}' = \text{cards} \setminus \{\text{card}?\}$

$\text{withdraw} \triangleq [\text{card}?: \text{cards}] \bullet \text{card}?.\text{withdraw}$

$\text{deposit} \triangleq [\text{card}?: \text{cards}] \bullet \text{card}?.\text{deposit}$

$\text{withdrawAvail} \triangleq [\text{card}?: \text{cards}] \bullet \text{card}?.\text{withdrawAvail}$

$\text{transferAvail} \triangleq [(\text{from}?, \text{to}?: \text{cards} \mid \text{from} \neq \text{to}) \bullet \text{from}?.\text{withdrawAvail}$

$\parallel \text{to}?.\text{deposit}$

Figura: CreditCards

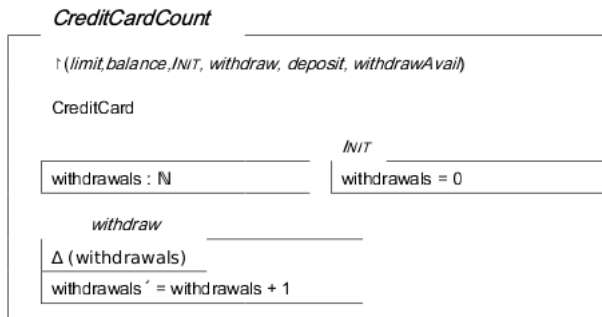


Figura: CreditCards



Tim G. Kimber

*Object-Z to Perfect Developer*

Chapter 2. Object-Z

Imperial College London, 2007