Desarrollo formal de Programas
Casos Refinamiento, 2007

Camilo Rueda [1]

[1] Universidad Javeriana-Cali, Colombia

PUJ 2007

Ascensores en formalismo de eventos

- Conjuntos (tipos):
  $PUERTA,\ ESTADO = \{abierta, cerrada\}$
- Constantes:
  $pisos \subseteq 1..n \land n \in \mathcal{N},\ ptas \subseteq PUERTA$
  $piso\_de\_pta \in ptas \rightarrow pisos$
- Variables:
  $estado\_pta$: estado de cada puerta
- Invariante abstracto:
  $estado\_pta \in ptas \rightarrow ESTADO$

Eventos

abrir=

ANY p **WHERE**

$p \in ptas$

THEN

$estado\_pta(p) := abierta$

END

cerrar=

ANY p **WHERE**

$p \in ptas \wedge estado\_pta(p) = abierta$

THEN

$estado\_pta(p) := cerrada$

END

Refinamiento(1)

Se ven los ascensores:

- Conjunto adicional (tipo): *ASCENSOR*.
- Constantes:
  $asc \subseteq ASCENSOR$, $asc\_de\_pta \in ptas \rightarrow asc$
- Variables: $asc\_en\_piso \in asc \rightarrow pisos$
  El piso en el que se encuentra cada ascensor

invariante de encadenamiento

- Una puerta solo está abierta si hay un ascensor allí

$$(\forall p \in ptas.\ estado\_pta(p) = abierta \Rightarrow (piso\_de\_pta(p) = asc\_en\_piso(asc\_de\_pta^{-1}(p))))$$

- Un ascensor no puede servir varias puertas del mismo piso

$$(\forall p_1, p_2 \in ptas\ .\ p_1 \neq p_1 \wedge piso\_de\_pta(p_1) = piso\_de\_pta(p_2) \Rightarrow asc\_de\_pta(p_1) \neq asc\_de\_pta(p_2))$$

Eventos(2)

abrir=

ANY p **WHERE**

$p \in ptas \wedge asc\_en\_piso(asc\_de\_pta(p)) = piso\_de\_pta(p)$

THEN

$estado\_pta(p) := abierta$

END

cerrar=

ANY p **WHERE**

$p \in ptas \wedge estado\_pta(p) = abierta$

THEN

$estado\_pta(p) := cerrada$

END

Eventos(2)

- Las puertas del piso en el que está el ascensor:
  $P_1 = piso\_de\_pta^{-1}(\{asc\_en\_piso(a)\})$
- Las puertas servidas por el ascensor a:
  $P_2 = asc\_de\_pta^{-1}(\{a\})$
- La puerta del ascensor del piso en el que se encuentra:
  $P_1 \cap P_2$

Cambiar_de_piso=
ANY a, p **WHERE**
$p \in pisos \wedge a \in asc \wedge p \neq asc\_en\_piso(a)$
$(estado\_pta(piso\_de\_pta^{-1}(\{asc\_en\_piso(a)\})$
$\cap\ asc\_de\_pta^{-1}(\{a\}))\ = \{cerrada\}$
THEN
$asc\_en\_piso(a) := p$
END

Refinamiento(3)

Se ve que el ascensor no da saltos:

- Variables: $destino \in asc \rightarrow pisos$
- Invariante de encadenamiento:

$$\boxed{\begin{array}{l}(\forall p \in ptas.\ estado\_pta(p) = abierta \\ \quad \Rightarrow \\ \quad destino(a) = piso\_de\_pta(p)\ )\end{array}}$$

Eventos(3)

Cambiar_de_piso1=

ANY a **WHERE**

$a \in asc \wedge asc\_en\_piso(a) < destino(a)$

$(estado\_pta(piso\_de\_pta^{-1}(\{asc\_en\_piso(a)\})$

$\cap\ asc\_de\_pta^{-1}(\{a\}))\ = \{cerrada\}$

THEN $asc\_en\_piso(a) := asc\_en\_piso(a) + 1$ **END**

Cambiar_de_piso2=

ANY a **WHERE**

$a \in asc \wedge asc\_en\_piso(a) > destino(a)$

$(estado\_pta(piso\_de\_pta^{-1}(\{asc\_en\_piso(a)\})$

$\cap\ asc\_de\_pta^{-1}(\{a\}))\ = \{cerrada\}$

THEN $asc\_en\_piso(a) := asc\_en\_piso(a) - 1$ **END**

Eventos(3)

Cambiar_destino=
ANY d, a **WHERE**
$a \in asc \wedge d \in pisos \wedge asc\_en\_piso(a) = destino(a)$
$(estado\_pta(piso\_de\_pta^{-1}(\{asc\_en\_piso(a)\})$
$\cap\ asc\_de\_pta^{-1}(\{a\}))\ = \{cerrada\}$
$d \neq destino(a)$
THEN $destino(a) = d$ **END**

abrir=
ANY p **WHERE**
$p \in ptas$
$\wedge asc\_en\_piso(asc\_de\_pta(p)) = piso\_de\_pta(p)$
$\wedge\ destino(asc\_de\_pta(p)) = piso\_de\_pta(p))$
THEN $estado\_pta(p) := abierta$ **END**

Refinamiento(4)

Botones en los pisos y en el ascensor:

- Variables:
  $boton\_piso \in pisos \rightarrow \{on, off\}$
  $boton\_asc \in asc \rightarrow (pisos \rightarrow \{on, off\})$
  $sched \in asc \rightarrow seq[pisos]$
- Invariante de encadenamiento:

$$
\begin{array}{l}
(\forall p \in pisos.\ boton\_piso(p) = on \quad \Rightarrow \\
\qquad (\exists a \in asc.\ p \in ran(sched(a))) \\
\wedge (\forall a \in asc \\
\qquad destino(a) = sched(a)(1) \\
\qquad \wedge\ (\forall p \in pisos\ .\ boton\_asc(a)(p) = on \\
\qquad\quad \Rightarrow p \in ran(sched(a))))
\end{array}
$$

Eventos(4)

opr_bot_piso1=
ANY p, a **WHERE**
$a \in asc \wedge p \in pisos \wedge boton\_piso(p) = off$
$p \notin ran(sched(a))$
THEN
$boton\_piso(p), sched(a) := on, sched(a) \leftarrow p$
END

opr_bot_piso2=
ANY p, a **WHERE**
$a \in asc \wedge p \in pisos \wedge boton\_piso(p) = off$
$p \in ran(sched(a))$
THEN
$boton\_piso(p) := on$
END

Eventos(5)

opr_bot_asc1=
ANY p, a **WHERE**
$a \in asc \wedge p \in pisos \wedge boton\_asc(a)(p) = off$
$p \notin ran(sched(a))$
THEN
$boton\_asc(a)(p), sched(a) := on, sched(a) \leftarrow p$
END

opr_bot_asc2=
ANY p, a **WHERE**
$a \in asc \wedge p \in pisos \wedge boton\_asc(a)(p) = off$
$p \in ran(sched(a))$
THEN
$boton\_asc(a)(p) := on$
END

Ejercicios

- Proponer nuevo refinamiento: Introducir los mensajes
  - Los ascensores y el piso envían mensajes a un controlador
  - El controlador envía mensajes a los ascensores (por ej. cambiar de piso) y al piso