# Composizione parallela

ParL) $\dfrac{p_1 \xrightarrow{\mu} q_1}{p_1|p_2 \xrightarrow{\mu} q_1|p_2}$ Com) $\dfrac{p_1 \xrightarrow{\lambda} q_1 \quad p_2 \xrightarrow{\overline{\lambda}} q_2}{p_1|p_2 \xrightarrow{\tau} q_1|q_2}$ ParR) $\dfrac{p_2 \xrightarrow{\mu} q_2}{p_1|p_2 \xrightarrow{\mu} p_1|q_2}$

i processi che girano in parallelo possono intrecciare le loro azioni o sincronizzarsi quando vengono eseguite due azioni

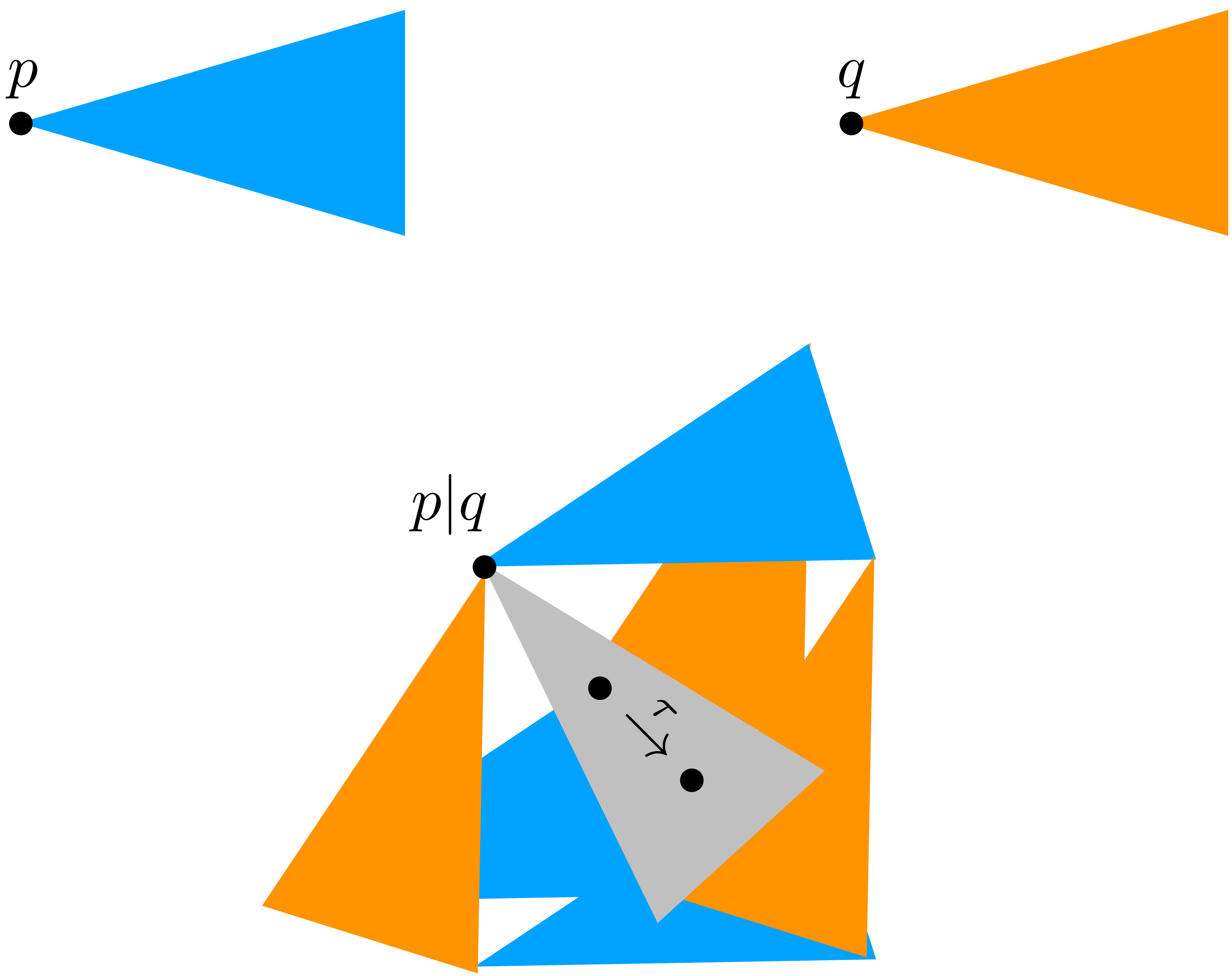
$P \triangleq \overline{coin}.coffee.\mathbf{nil}$ $\qquad M \triangleq coin.(\overline{coffee}.\mathbf{nil} + \overline{tea}.\mathbf{nil})$

$$P|M \xrightarrow{\overline{coin}} coffee.\mathbf{nil}|M$$

$$P|M \xrightarrow{coin} P|(\overline{coffee}.\mathbf{nil} + \overline{tea}.\mathbf{nil})$$

$$P|M \xrightarrow{\tau} coffee.\mathbf{nil}|(\overline{coffee}.\mathbf{nil} + \overline{tea}.\mathbf{nil})$$

# LTS del processo



# CCS: buffer paralleli

$$B_0^1 | B_0^1$$

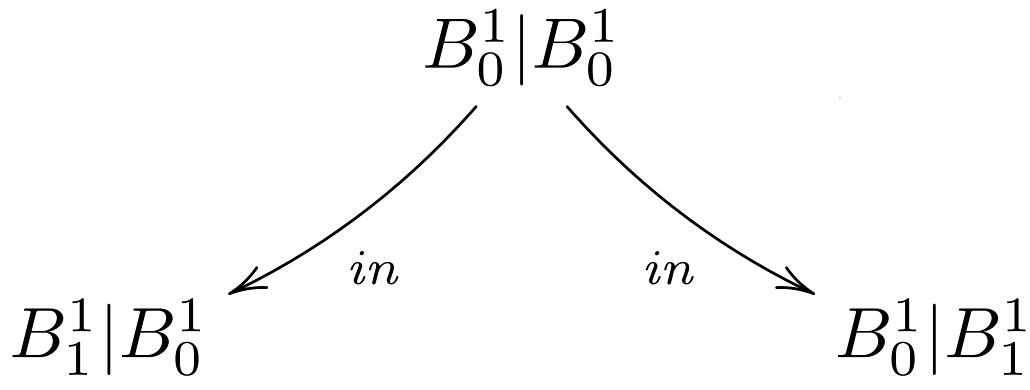
$$B_0^1 \triangleq in.B_1^1$$

$$B_1^1 \triangleq \overline{out}.B_0^1$$

# CCS: buffer paralleli

$$B_0^1 \triangleq in.B_1^1$$

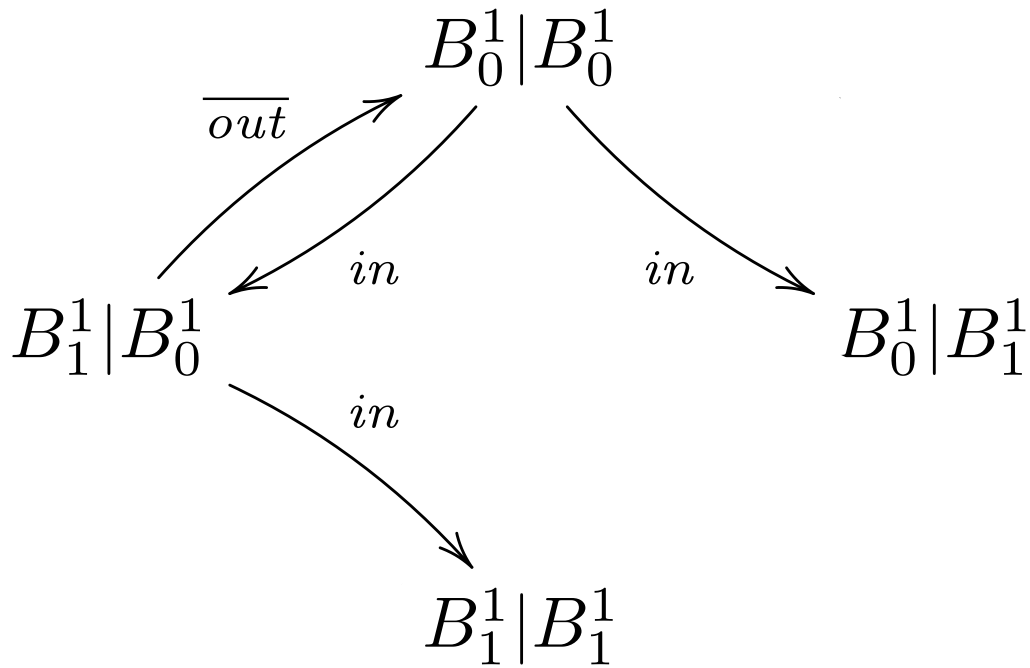
$$B_1^1 \triangleq \overline{out}.B_0^1$$



# CCS: buffer paralleli

$$B_0^1 \triangleq in.B_1^1$$

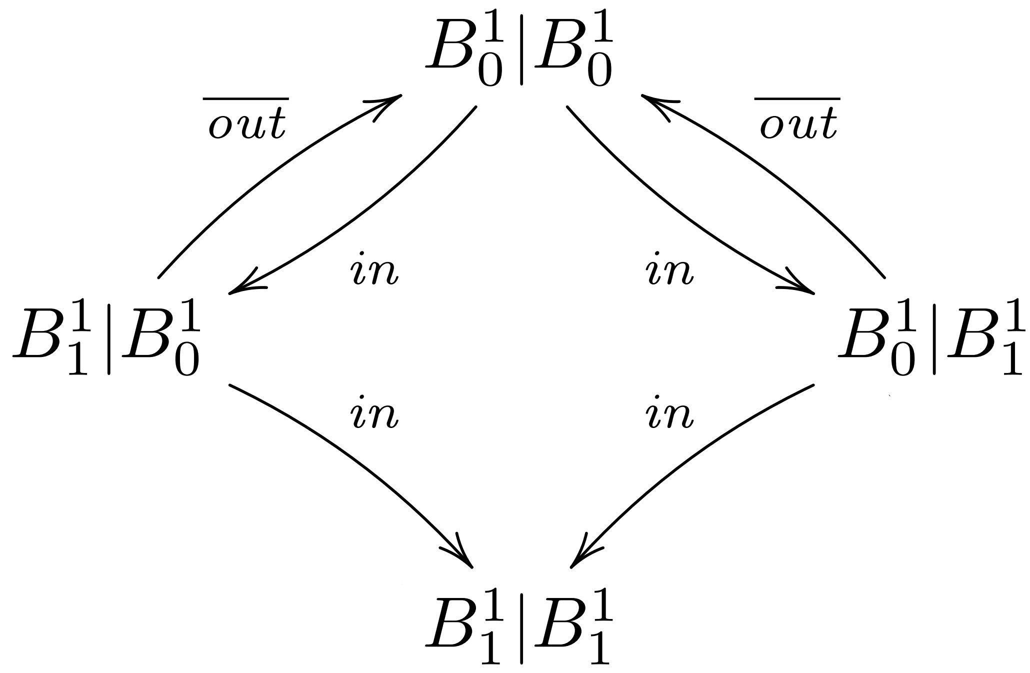
$$B_1^1 \triangleq \overline{out}.B_0^1$$



# CCS: buffer paralleli

$B_0^1 \triangleq in.B_1^1$

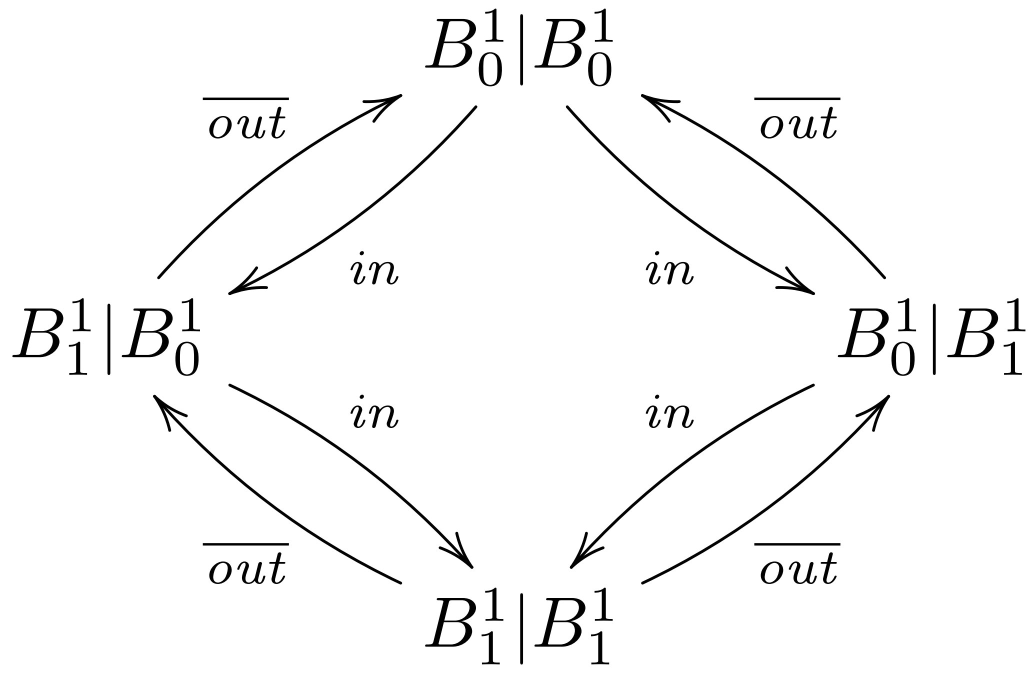
$B_1^1 \triangleq \overline{out}.B_0^1$

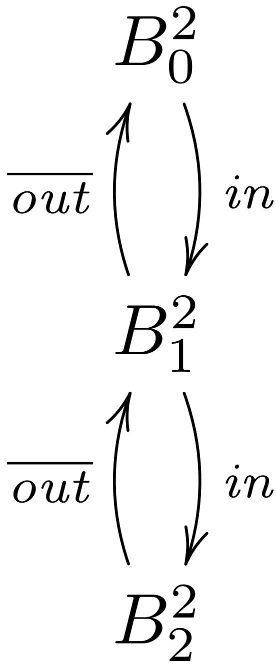


# CCS: buffer paralleli

$$B^1_0 \triangleq in.B^1_1$$

$$B^1_1 \triangleq \overline{out}.B^1_0$$




confrontare con il buffer a capacità 2

# Restrizione

Res) $$\frac{p \xrightarrow{\mu} q \quad \mu \notin \{\alpha, \overline{\alpha}\}}{p\backslash\alpha \xrightarrow{\mu} q\backslash\alpha}$$

rende il canale $\alpha$ privato a $p$

nessuna interazione sul canale $\alpha$ con l'ambiente

se $p$ è la composizione parallela dei processi, allora possono sincronizzarsi su $\alpha$

$P \triangleq \overline{coin}.coffee.\mathbf{nil}$ $\qquad M \triangleq coin.(\overline{coffee}.\mathbf{nil} + \overline{tea}.\mathbf{nil})$

$$(P|M)\backslash coin\backslash coffee\backslash tea \xrightarrow{\tau} (coffee.\mathbf{nil}|\overline{coffee}.\mathbf{nil} + \overline{tea}.\mathbf{nil})\backslash coin\backslash coffee\backslash tea$$

$$(coffee.\mathbf{nil}|\overline{coffee}.\mathbf{nil} + \overline{tea}.\mathbf{nil})\backslash coin\backslash coffee\backslash tea \xrightarrow{\tau} (\mathbf{nil}|\mathbf{nil})\backslash coin\backslash coffee\backslash tea$$

# Restrizione: shorthand

dato $S = \{\alpha_1, ..., \alpha_n\}$ scriviamo $p \backslash S$

invece di $p \backslash \alpha_1 \ldots \backslash \alpha_n$

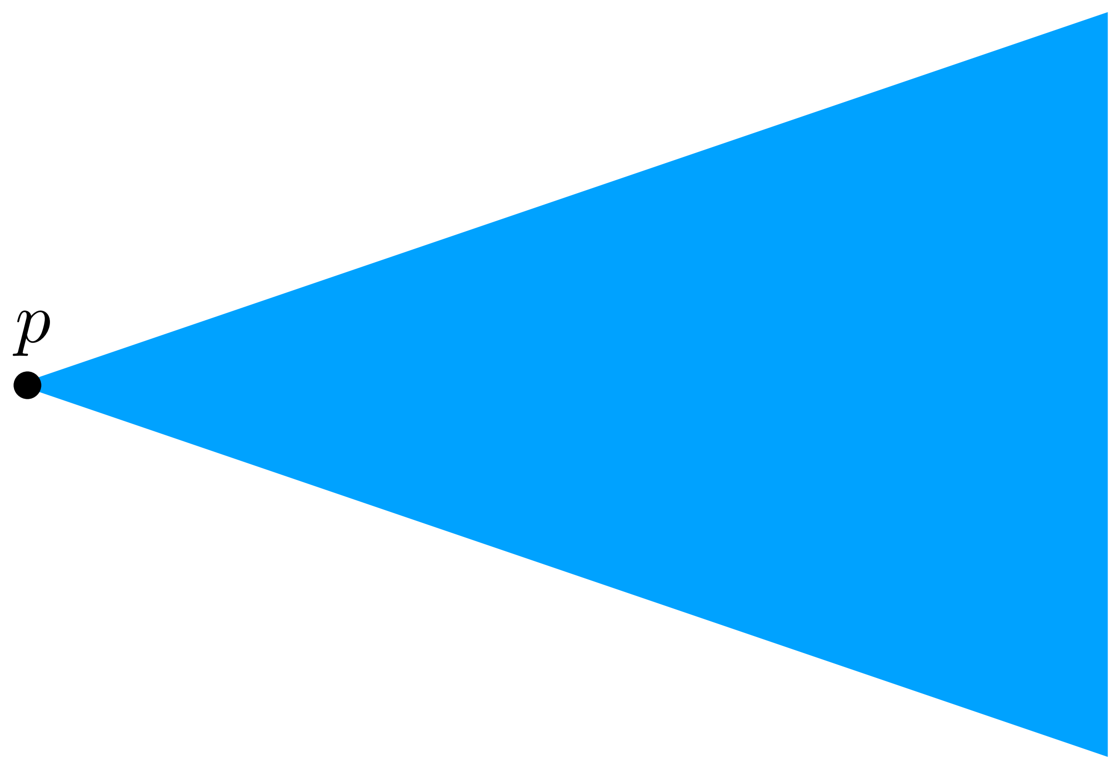
omettiamo $\mathbf{nil}$

$$P \triangleq \overline{coin}.coffee \qquad M \triangleq coin.(\overline{coffee} + \overline{tea})$$

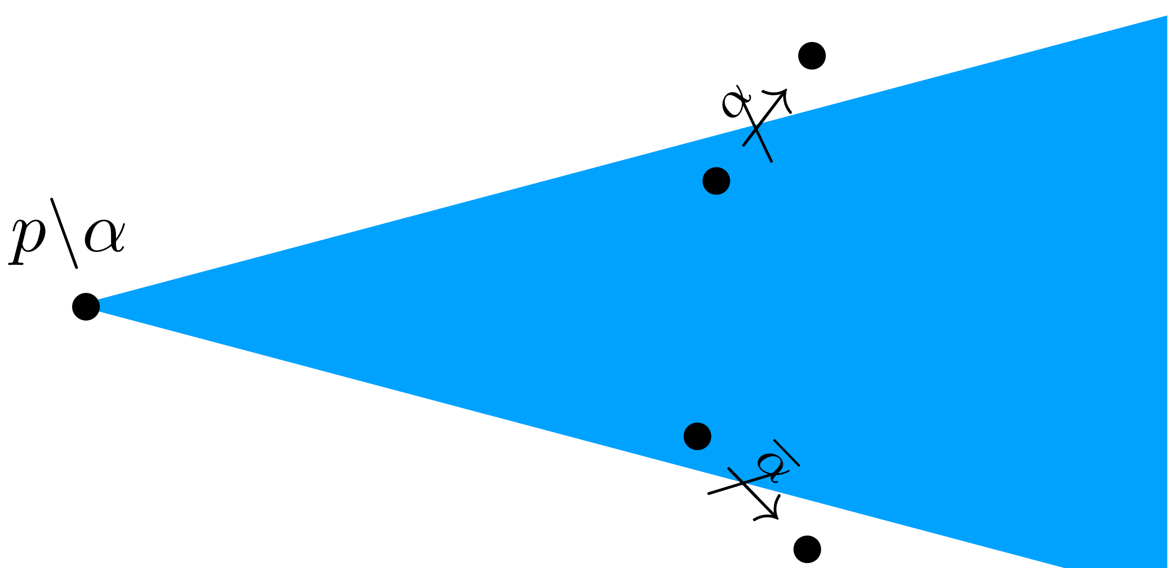
$$S \triangleq \{coin, coffee, tea\}$$

$$(P|M)\backslash S \xrightarrow{\tau} (coffee|\overline{coffee} + \overline{tea})\backslash S \xrightarrow{\tau} (\mathbf{nil}|\mathbf{nil})\backslash S$$

# LTS del processo



# LTS del processo



# ridenominazione

$$\text{Rel)}\ \frac{p \xrightarrow{\mu} q}{p[\phi] \xrightarrow{\phi(\mu)} q[\phi]}$$

rinomina i canali di un'azione secondo $\phi$

assumiamo $\phi(\tau) = \tau$ $\phi(\overline{\lambda}) = \overline{\phi(\lambda)}$

ci permette di riusare i processi

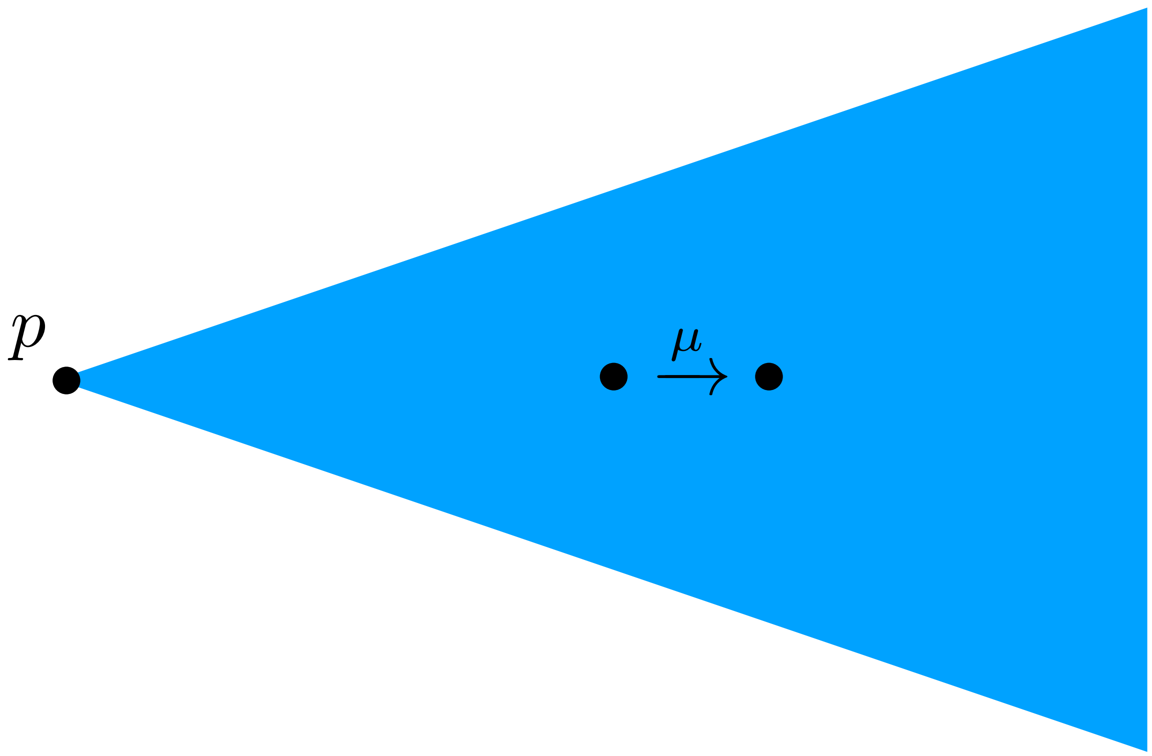
$P \triangleq \overline{coin}.coffee$

$\phi(coin) = moneta$

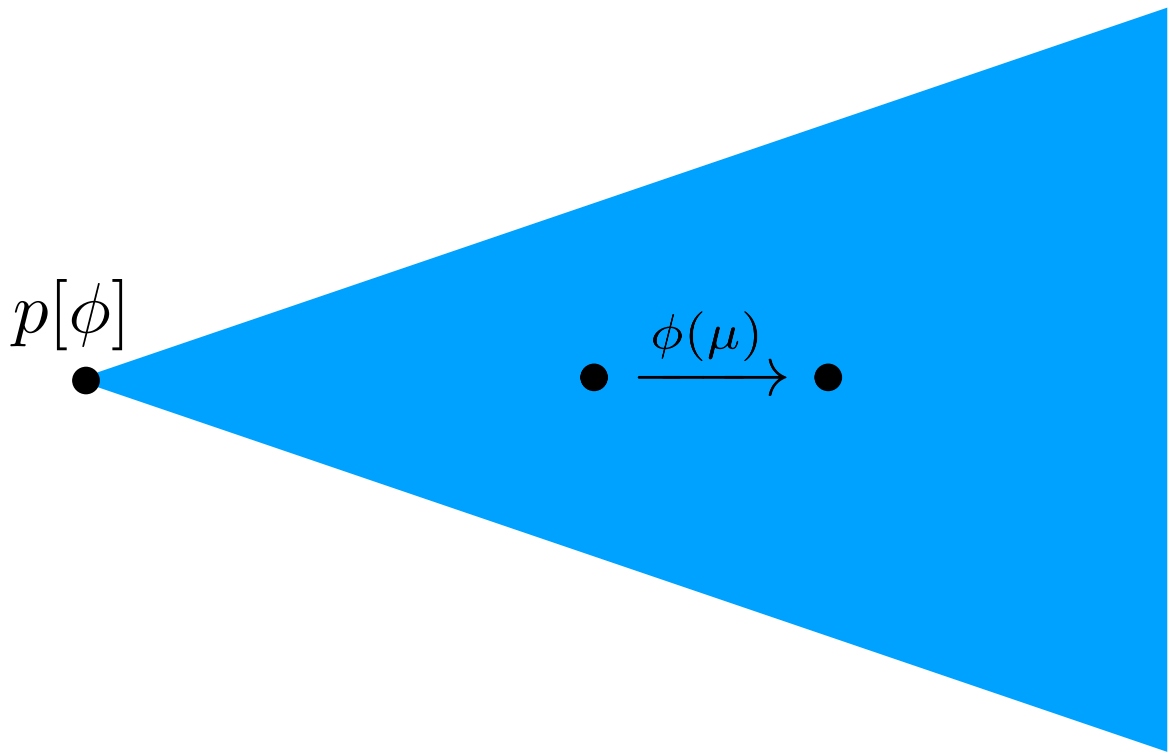
$\phi(coffee) = caffè$

$$P[\phi] \xrightarrow{\overline{moneta}} coffee[\phi] \xrightarrow{caffè} \mathbf{nil}[\phi]$$

# LTS del processo



# LTS del processo



# CCS semantica operazionale

Act) $$\frac{}{\mu.p \xrightarrow{\mu} p}$$

Res) $$\frac{p \xrightarrow{\mu} q \quad \mu \notin \{\alpha, \overline{\alpha}\}}{p \backslash \alpha \xrightarrow{\mu} q \backslash \alpha}$$

Rel) $$\frac{p \xrightarrow{\mu} q}{p[\phi] \xrightarrow{\phi(\mu)} q[\phi]}$$

SumL) $$\frac{p_1 \xrightarrow{\mu} q}{p_1 + p_2 \xrightarrow{\mu} q}$$

SumR) $$\frac{p_2 \xrightarrow{\mu} q}{p_1 + p_2 \xrightarrow{\mu} q}$$

ParL) $$\frac{p_1 \xrightarrow{\mu} q_1}{p_1 | p_2 \xrightarrow{\mu} q_1 | p_2}$$

Com) $$\frac{p_1 \xrightarrow{\lambda} q_1 \quad p_2 \xrightarrow{\overline{\lambda}} q_2}{p_1 | p_2 \xrightarrow{\tau} q_1 | q_2}$$

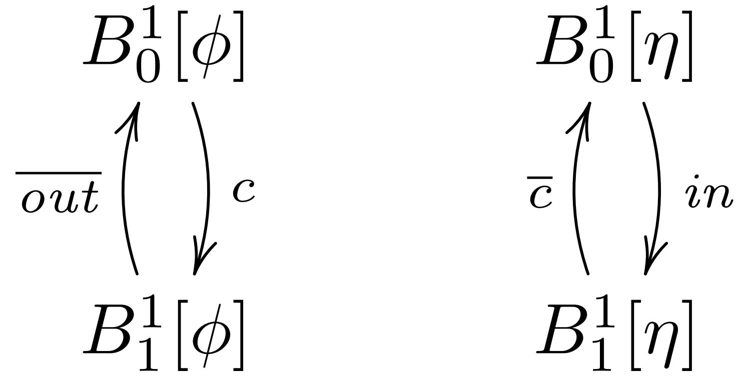
ParR) $$\frac{p_2 \xrightarrow{\mu} q_2}{p_1 | p_2 \xrightarrow{\mu} p_1 | q_2}$$

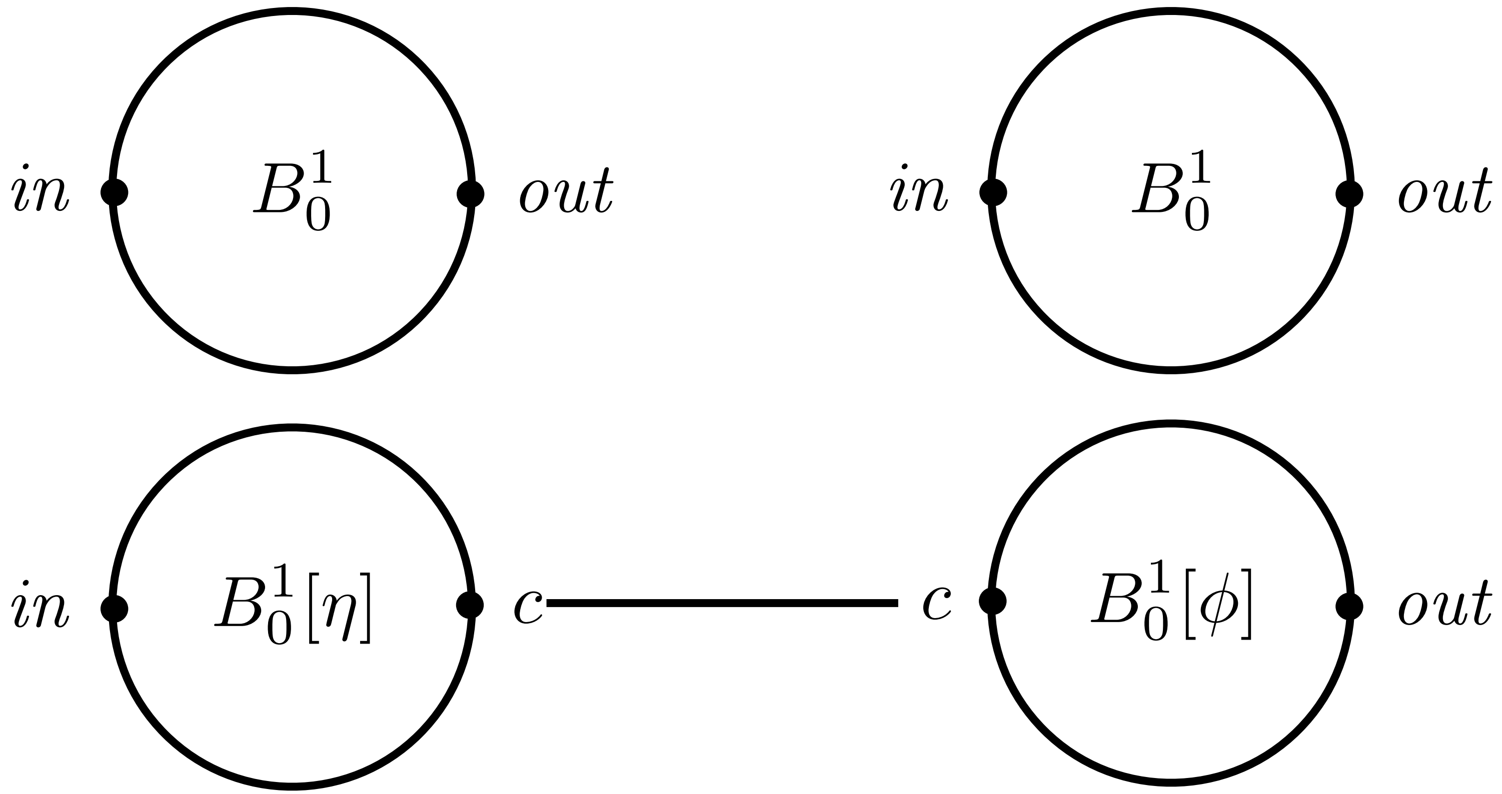
Rec) $$\frac{p[{}^{\mathbf{rec}\ x.\ p}/_{x}] \xrightarrow{\mu} q}{\mathbf{rec}\ x.\ p \xrightarrow{\mu} q}$$

# Buffer collegati

$B_0^1 \triangleq in.B_1^1$ $\qquad \eta(out) = c$

$B_1^1 \triangleq \overline{out}.B_0^1$ $\qquad \phi(in) = c$

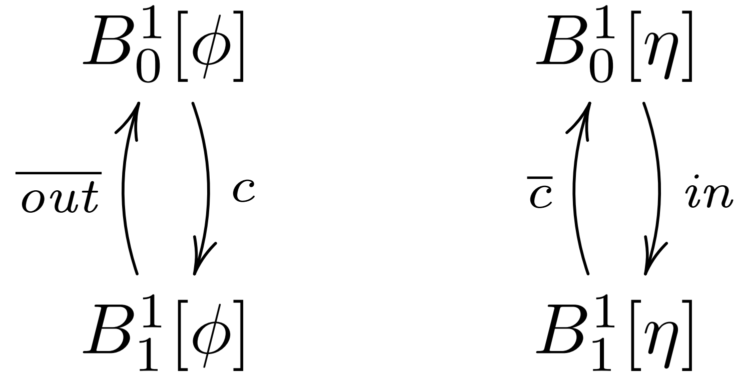


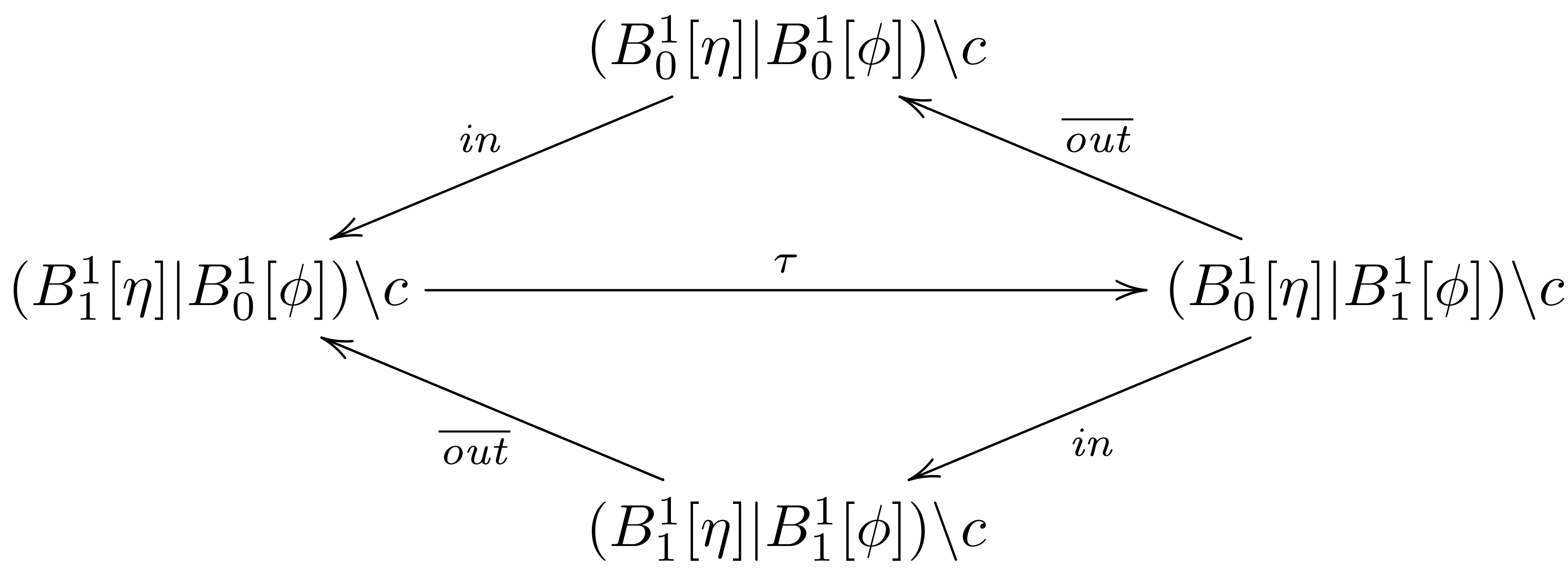



# Buffer collegati

$B_0^1 \triangleq in.B_1^1$ $\qquad \eta(out) = c$

$B_1^1 \triangleq \overline{out}.B_0^1$ $\qquad \phi(in) = c$



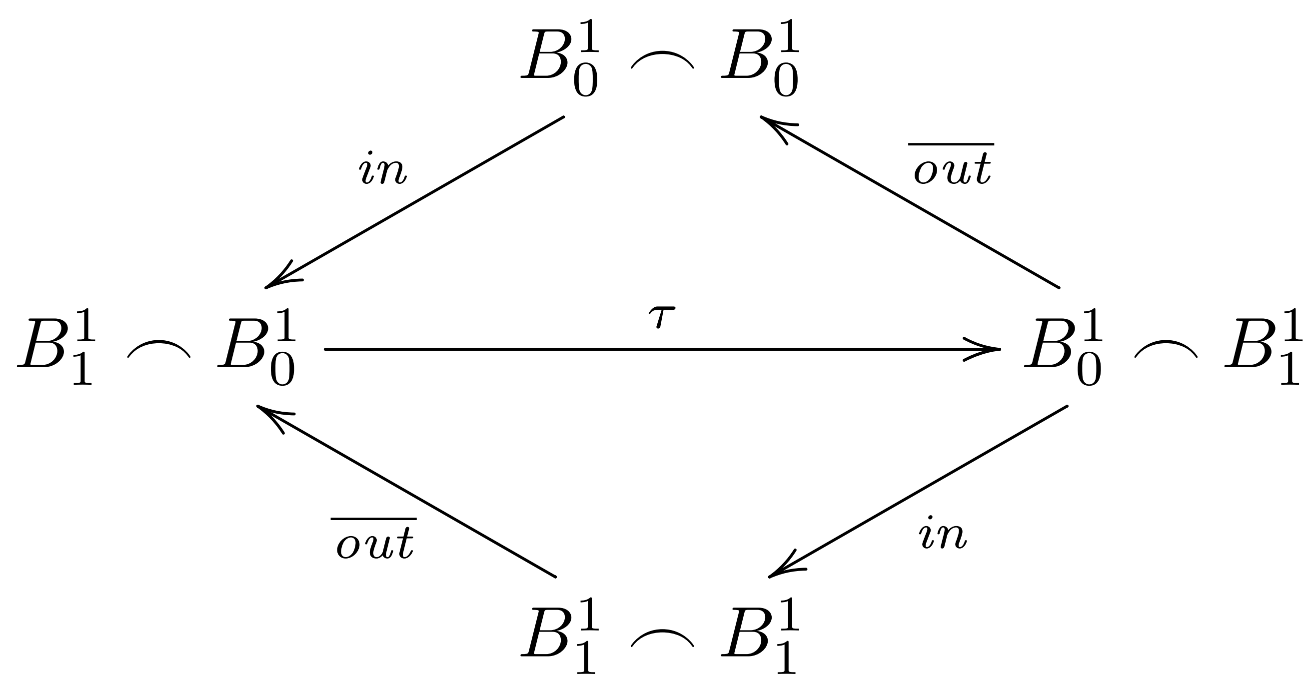



# Buffer collegati

$B_0^1 \triangleq in.B_1^1 \qquad \eta(out) = c$

$B_1^1 \triangleq \overline{out}.B_0^1 \qquad \phi(in) = c$

$$p \frown q \triangleq (p[\eta] | q[\phi]) \backslash c$$



# Buffer collegati booleani

$$B_\emptyset \triangleq in_t.B_t + in_f.B_f \qquad \eta(out_t) = c_t \qquad \phi(in_t) = c_t$$

$$B_t \triangleq \overline{out}_t.B_\emptyset \qquad \eta(out_f) = c_f \qquad \phi(in_f) = c_f$$

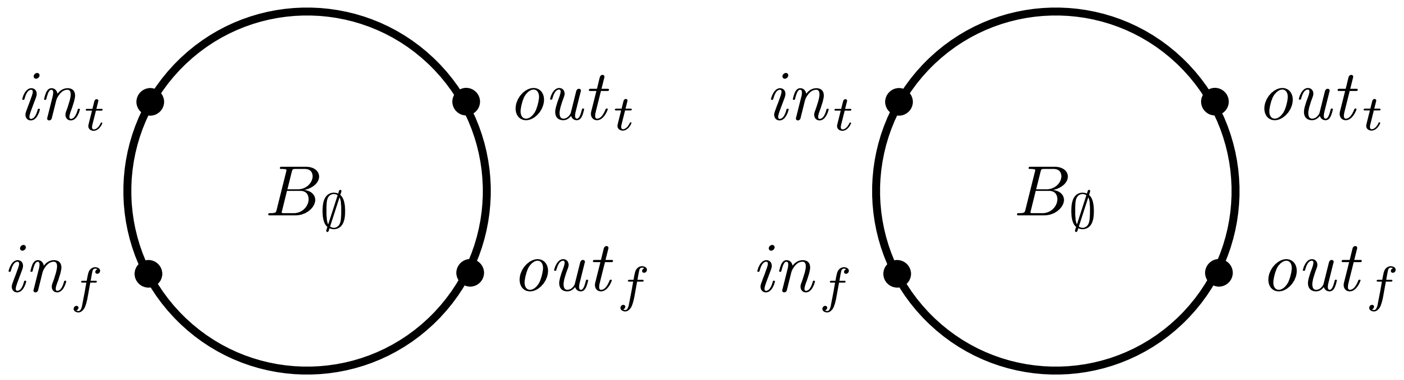
$$B_f \triangleq \overline{out}_f.B_\emptyset \qquad p \frown q \triangleq (p[\eta] | q[\phi]) \backslash \{c_t, c_f\}$$

# Buffer collegati booleani

$$B_\emptyset \triangleq in_t.B_t + in_f.B_f$$

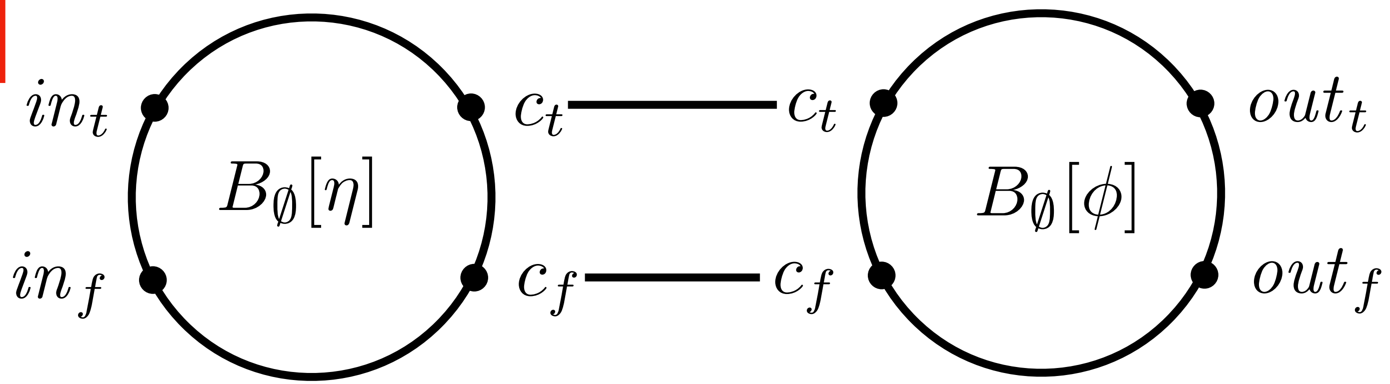
$$B_t \triangleq \overline{out_t}.B_\emptyset$$

$$B_f \triangleq \overline{out_f}.B_\emptyset$$




$\eta(out_t) = c_t \quad \phi(in_t) = c_t$

$\eta(out_f) = c_f \quad \phi(in_f) = c_f$

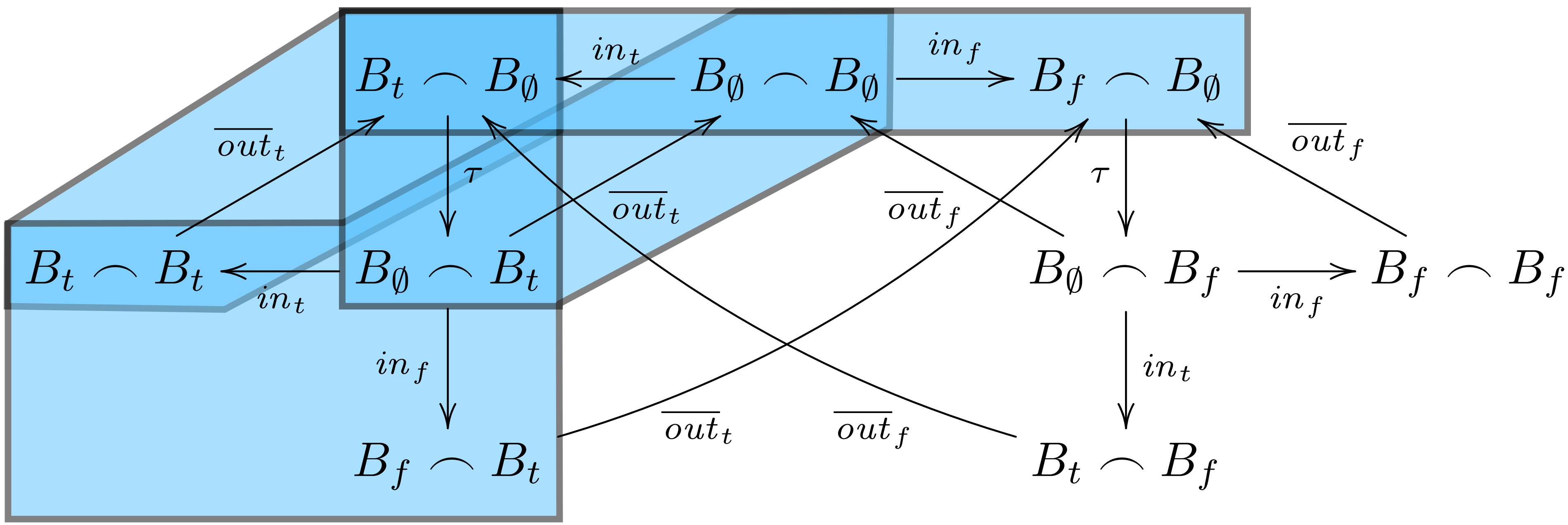


# Buffer collegati booleani

$$B_\emptyset \triangleq in_t.B_t + in_f.B_f \qquad \eta(out_t) = c_t \qquad \phi(in_t) = c_t$$

$$B_t \triangleq \overline{out}_t.B_\emptyset \qquad \eta(out_f) = c_f \qquad \phi(in_f) = c_f$$

$$B_f \triangleq \overline{out}_f.B_\emptyset \qquad p \frown q \triangleq (p[\eta]|q[\phi])\backslash\{c_t, c_f\}$$

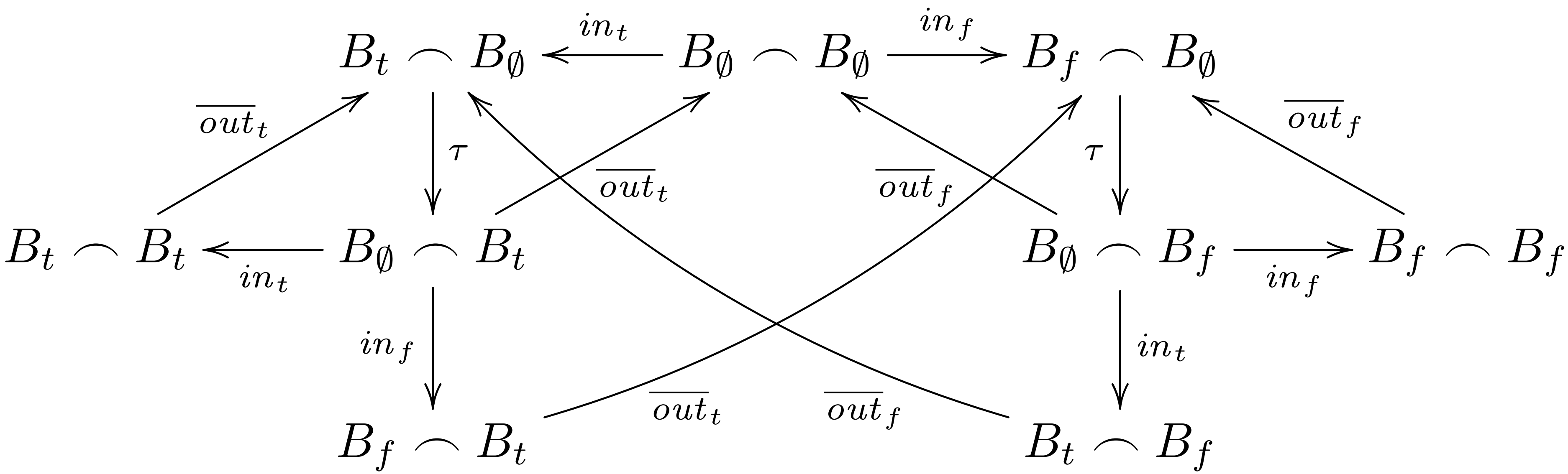


# Buffer collegati booleani

$B_\emptyset \triangleq in_t.B_t + in_f.B_f$ $\qquad \eta(out_t) = c_t \qquad \phi(in_t) = c_t$

$B_t \triangleq \overline{out}_t.B_\emptyset$ $\qquad \eta(out_f) = c_f \qquad \phi(in_f) = c_f$

$B_f \triangleq \overline{out}_f.B_\emptyset$ $\qquad p \frown q \triangleq (p[\eta] | q[\phi])\backslash\{c_t, c_f\}$



# Sono uguali questi processi?

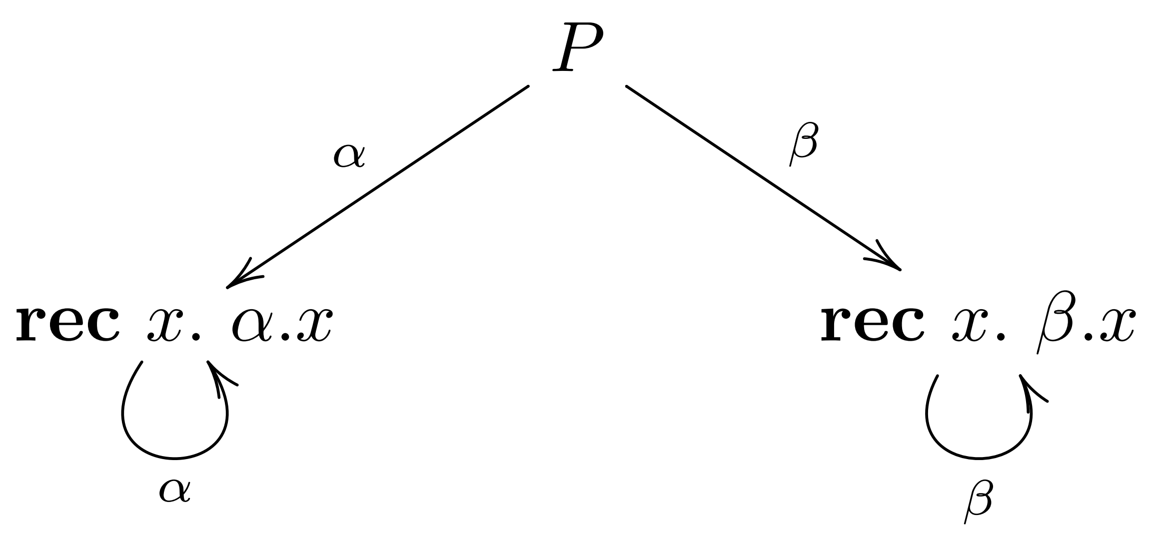
$$P \triangleq (\mathbf{rec}\ x.\ \alpha.x) + (\mathbf{rec}\ x.\ \beta.x)$$

$$Q \triangleq \mathbf{rec}\ x.\ (\alpha.x + \beta.x)$$

$$R \triangleq \mathbf{rec}\ x.\ (\alpha.x + \beta.\mathbf{nil})$$

# Esercizio: LTS?

$$P \triangleq (\mathbf{rec}\ x.\ \alpha.x) + (\mathbf{rec}\ x.\ \beta.x)$$

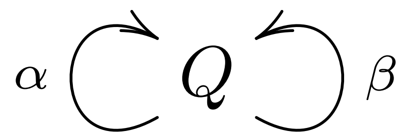


# Esercizio: LTS?

$$Q \triangleq \mathbf{rec}\ x.\ (\alpha.x + \beta.x)$$

$$Q \triangleq \mathbf{rec}\ x.\ \alpha.x + \beta.x$$

$$Q \triangleq \alpha.Q + \beta.Q$$

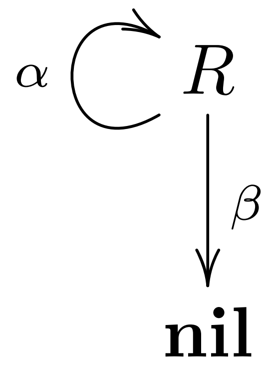


# Esercizio: LTS?

$$R \triangleq \mathbf{rec}\ x.\ (\alpha.x + \beta.\mathbf{nil})$$

$$R \triangleq \mathbf{rec}\ x.\ \alpha.x + \beta$$

$$R \triangleq \alpha.R + \beta$$



# Somma vs parallelismo

$$R \triangleq \mathbf{rec}\ x.\ (\alpha.x + \beta.\mathbf{nil})$$

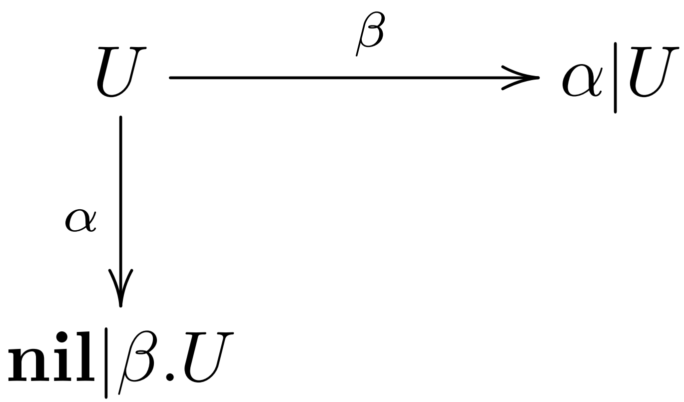
$$U \triangleq \mathbf{rec}\ x.\ ((\alpha.\mathbf{nil})|\beta.x)$$

# Esercizio: LTS?

$$U \triangleq \mathbf{rec}\ x.\ ((\alpha.\mathbf{nil})|\beta.x)$$

$$U \triangleq \mathbf{rec}\ x.\ \alpha|\beta.x$$

$$U \triangleq \alpha|\beta.U$$

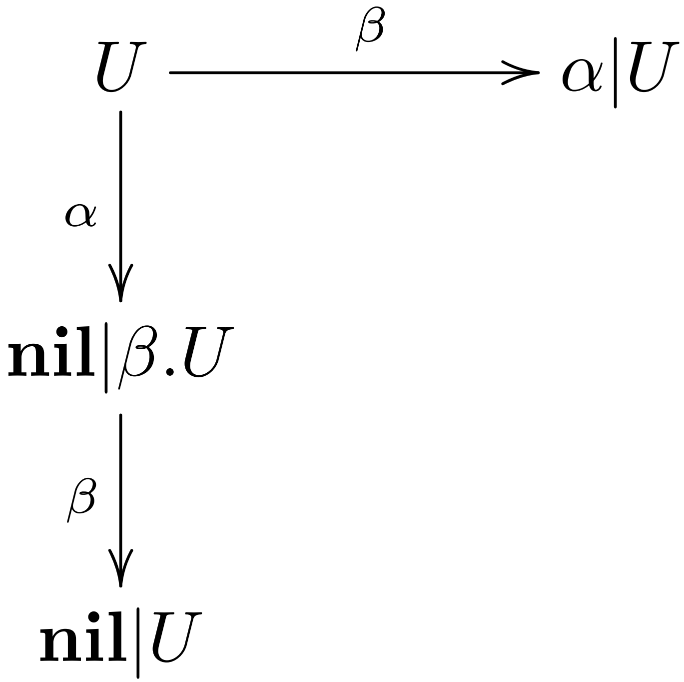


# Esercizio: LTS?

$$U \triangleq \mathbf{rec}\ x.\ ((\alpha.\mathbf{nil})|\beta.x)$$

$$U \triangleq \mathbf{rec}\ x.\ \alpha|\beta.x$$

$$U \triangleq \alpha|\beta.U$$

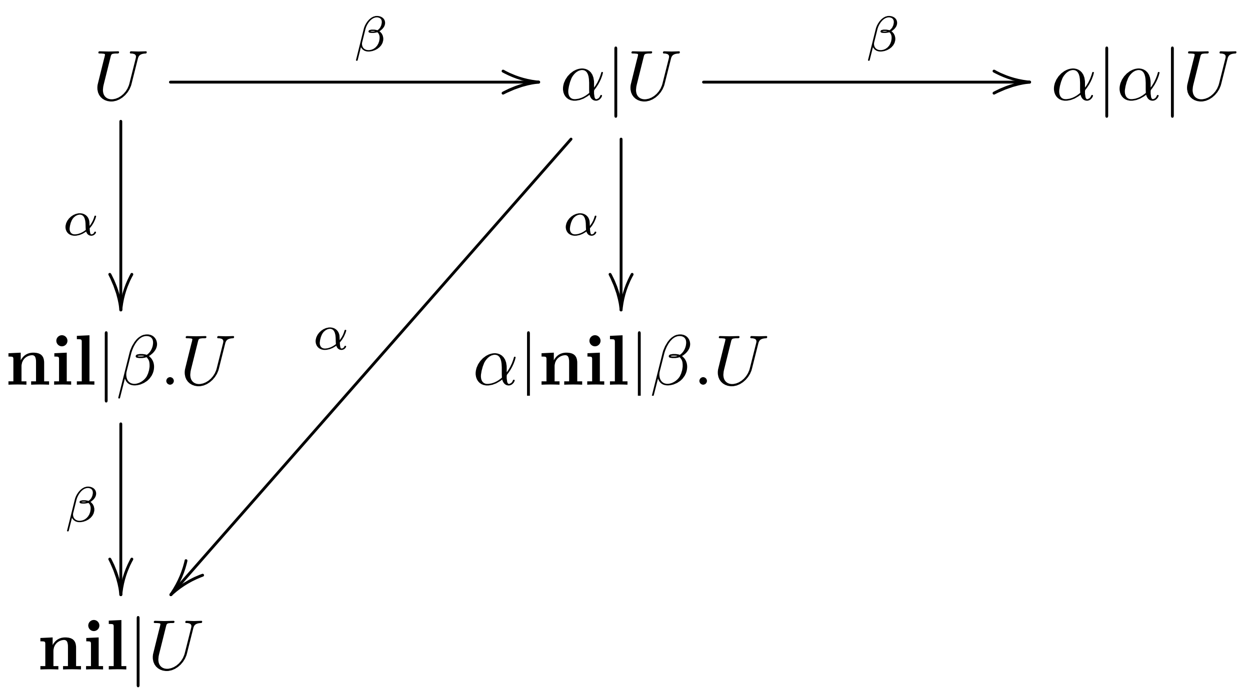


# Esercizio: LTS?

$$U \triangleq \mathbf{rec}\ x.\ ((\alpha.\mathbf{nil})|\beta.x)$$

$$U \triangleq \mathbf{rec}\ x.\ \alpha|\beta.x$$

$$U \triangleq \alpha|\beta.U$$

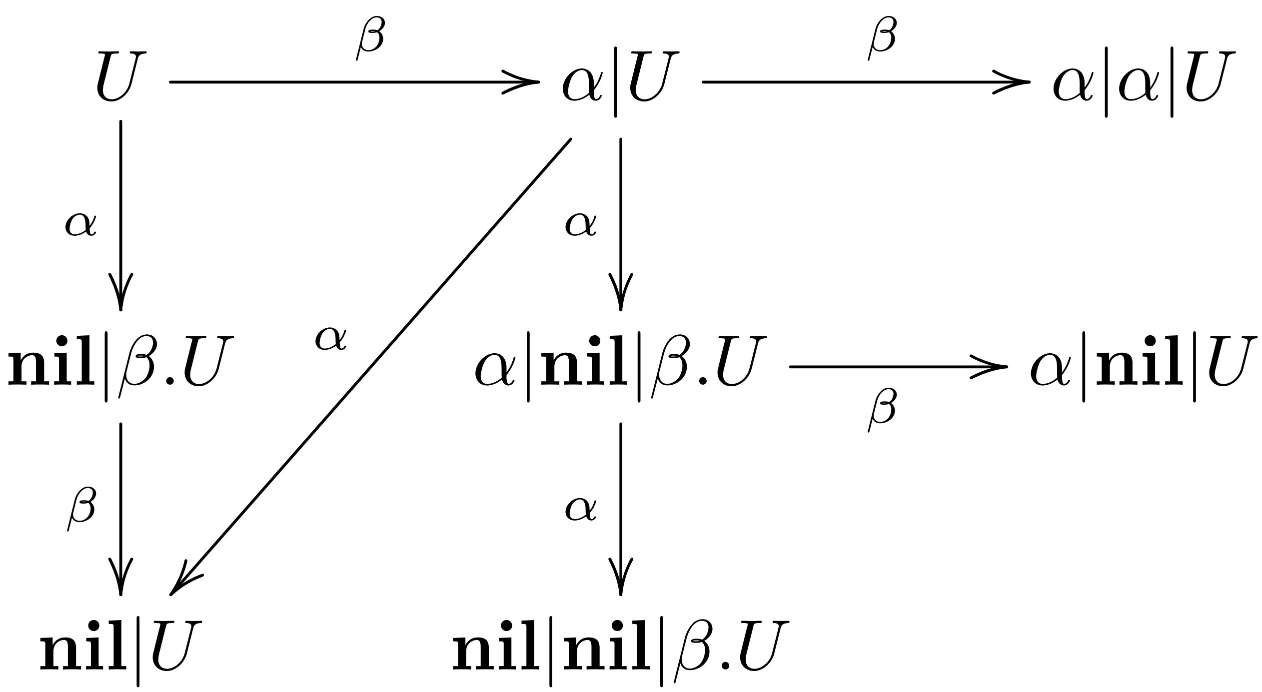


# Esercizio: LTS?

$$U \triangleq \mathbf{rec}\ x.\ ((\alpha.\mathbf{nil})|\beta.x)$$

$$U \triangleq \mathbf{rec}\ x.\ \alpha|\beta.x$$

$$U \triangleq \alpha|\beta.U$$

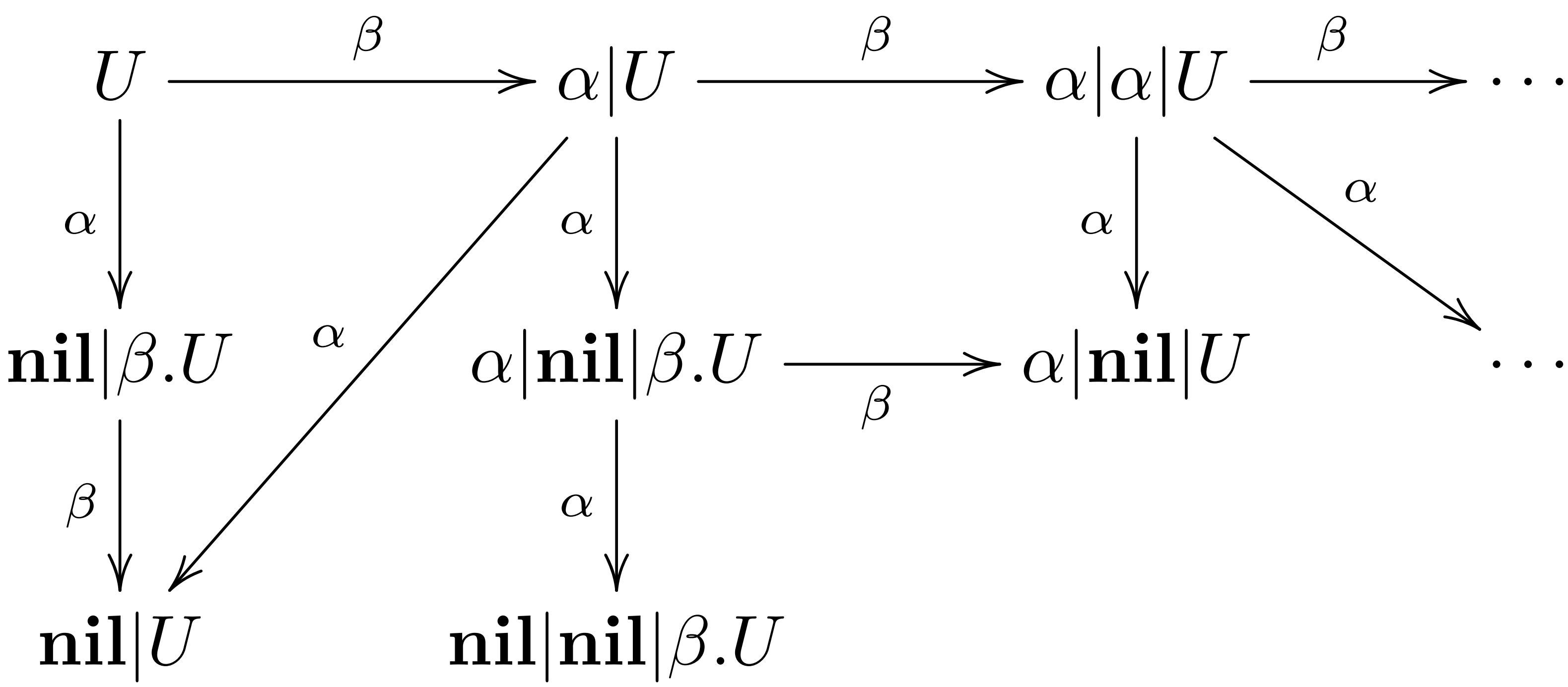


# Esercizio: LTS?

$$U \triangleq \mathbf{rec}\ x.\ ((\alpha.\mathbf{nil})|\beta.x)$$

$$U \triangleq \mathbf{rec}\ x.\ \alpha|\beta.x$$

$$U \triangleq \alpha|\beta.U$$

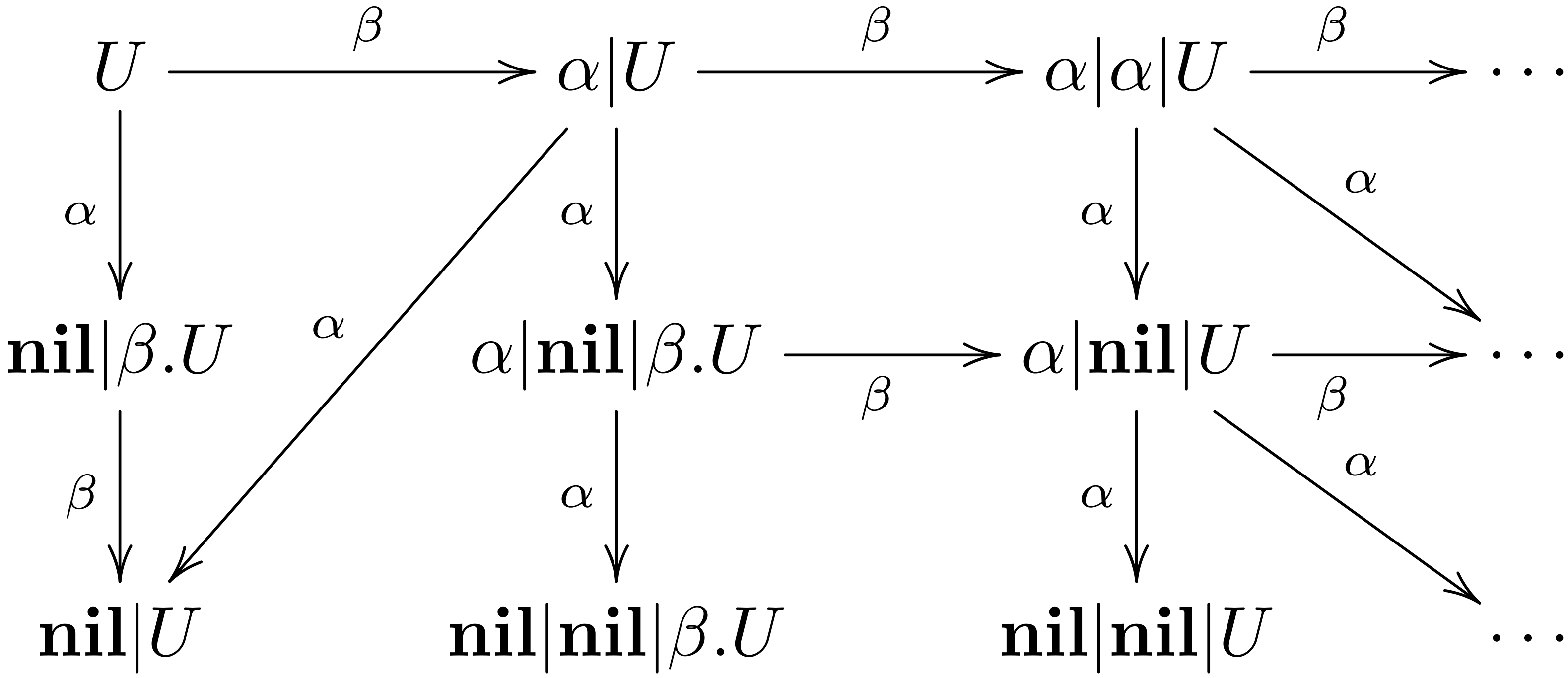


# Esercizio: LTS?

$$U \triangleq \mathbf{rec}\ x.\ ((\alpha.\mathbf{nil})|\beta.x)$$

$$U \triangleq \mathbf{rec}\ x.\ \alpha|\beta.x$$

$$U \triangleq \alpha|\beta.U$$



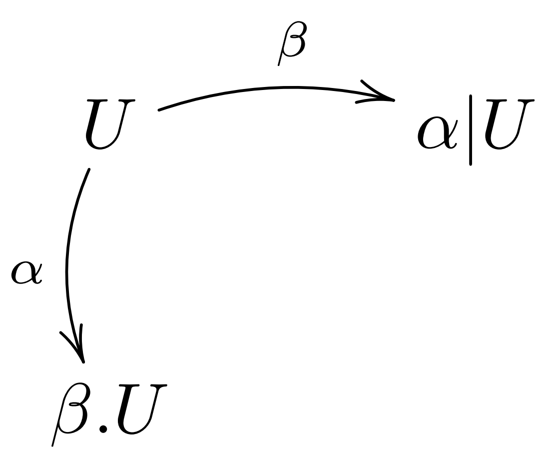
# Esercizio: LTS?

ignoriamo **nil**

$$U \triangleq \mathbf{rec}\ x.\ ((\alpha.\mathbf{nil})|\beta.x)$$

$$U \triangleq \mathbf{rec}\ x.\ \alpha|\beta.x$$

$$U \triangleq \alpha|\beta.U$$



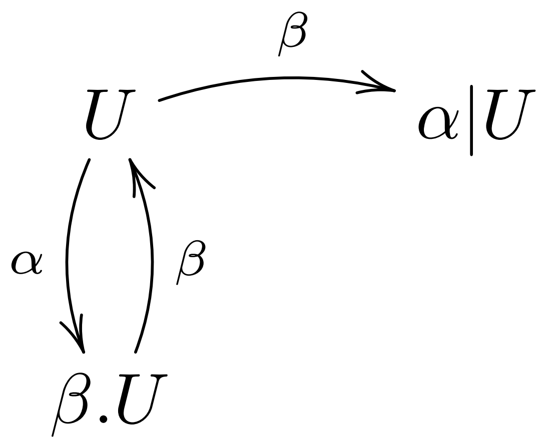
# Esercizio: LTS?

ignoriamo **nil**

$$U \triangleq \mathbf{rec}\ x.\ ((\alpha.\mathbf{nil})|\beta.x)$$

$$U \triangleq \mathbf{rec}\ x.\ \alpha|\beta.x$$

$$U \triangleq \alpha|\beta.U$$



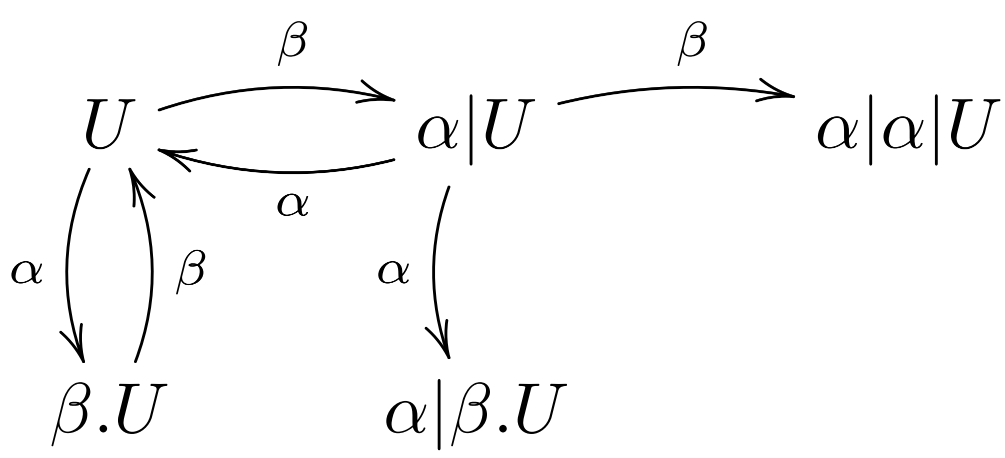
# Esercizio: LTS?

ignoriamo **nil**

$$U \triangleq \mathbf{rec}\ x.\ ((\alpha.\mathbf{nil})|\beta.x)$$

$$U \triangleq \mathbf{rec}\ x.\ \alpha|\beta.x$$

$$U \triangleq \alpha|\beta.U$$



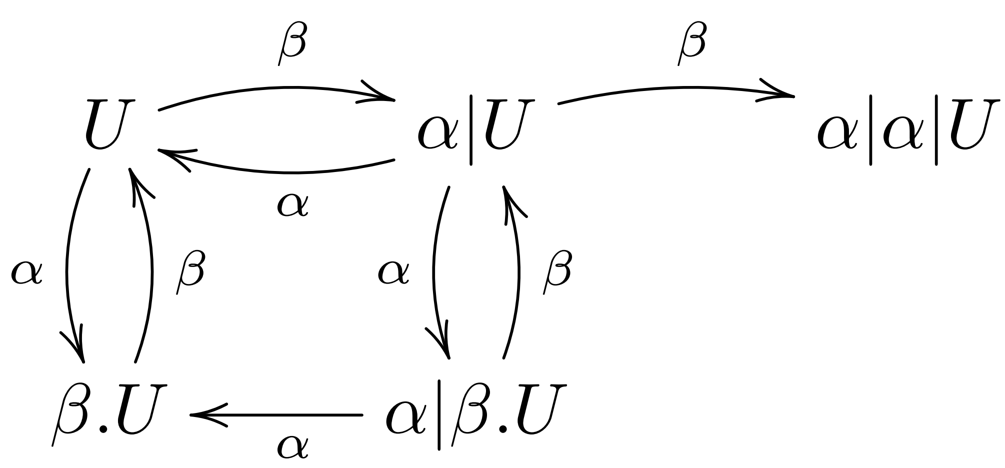
# Esercizio: LTS?

ignoriamo **nil**

$$U \triangleq \mathbf{rec}\ x.\ ((\alpha.\mathbf{nil})|\beta.x)$$

$$U \triangleq \mathbf{rec}\ x.\ \alpha|\beta.x$$

$$U \triangleq \alpha|\beta.U$$



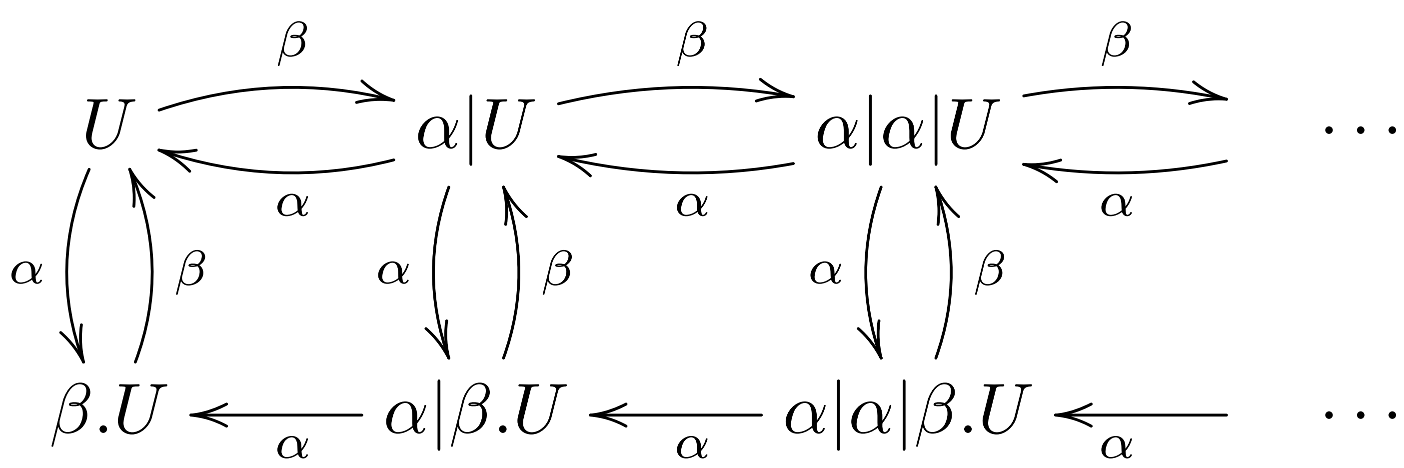
# Esercizio: LTS?

ignoriamo **nil**

$$U \triangleq \mathbf{rec}\ x.\ ((\alpha.\mathbf{nil})|\beta.x)$$

$$U \triangleq \mathbf{rec}\ x.\ \alpha|\beta.x$$

$$U \triangleq \alpha|\beta.U$$



# Esercizio (da consegnare)

Scrivere un contatore interattivo modulo 4 in CCS

Il processo contatore ha quattro canali di ingresso:

*inc*, *val*, *reset*, *stop*

e quattro canali di uscita:

$c_0$, $c_1$, $c_2$, $c_3$

usato per visualizzare il valore corrente del contatore

Disegna l'LTS del processo contatore.