PF transform: conditions, coreflexives and design by contract

J.N. Oliveira

Dept. Informática,
Universidade do Minho
Braga, Portugal

DI/UM, 2007 (last update: Nov. 2014)

Recall

Some basic rules of the PF-transform:

ϕ	$PF\ \phi$
$\langle \exists\ a\ ::\ b\ R\ a \wedge a\ S\ c \rangle$	$b(R \cdot S)c$
$\langle \forall\ a, b\ ::\ b\ R\ a \Rightarrow b\ S\ a \rangle$	$R \subseteq S$
$\langle \forall\ a\ ::\ a\ R\ a \rangle$	$id \subseteq R$
$b\ R\ a \wedge c\ S\ a$	$(b, c)\langle R, S \rangle a$
$b\ R\ a \wedge d\ S\ c$	$(b, d)(R \times S)(a, c)$
$b\ R\ a \wedge b\ S\ a$	$b\ (R \cap S)\ a$
$b\ R\ a \vee b\ S\ a$	$b\ (R \cup S)\ a$
$(f\ b)\ R\ (g\ a)$	$b(f^{\circ} \cdot R \cdot g)a$
TRUE	$b\ \top\ a$
FALSE	$b\ \bot\ a$

Question

- The PF-transform seems applicable to transforming **binary** predicates only, easily converted to binary relations, eg.

$$\phi(y, x) \triangleq \; y - 1 = 2x$$

which transforms to function $y = 2x + 1$, etc.

- What about transforming predicates such as the following

$$\langle \forall \; x, y \;\; : \;\; y = \mathit{twice} \; x \wedge \mathit{even} \; x : \;\; \mathit{even} \; y \rangle \tag{141}$$

expressing the fact that function $\mathit{twice} \; x \triangleq \; 2x$ preserves even numbers, where $\mathit{even} \; x \triangleq \; \mathit{rem}(x, 2) = 0$ is a **unary** predicate?

Observation

- As already noted, (141) is a proposition stating that function *twice* **preserves** even numbers.
- In general, a function $A \xleftarrow{f} A$ is said to **preserve** a given predicate ϕ iff the following holds:

$$\langle \forall\ x\ :\ \phi\ x :\ \phi\ (f\ x) \rangle \tag{142}$$

- Proposition (142) itself is a particular case of

$$\langle \forall\ x\ :\ \phi\ x :\ \psi\ (f\ x) \rangle \tag{143}$$

which states that f **ensures** property ψ on its **output** every time property ϕ holds on its **input**.

Answer

We first PF-transform the scope of the quantification:

$$
\begin{array}{cl}
& y = twice\ x \wedge even\ x \\
\equiv & \qquad \{ \text{ introduce } z \text{ by } \exists\text{-one-point (15) } \} \\
& \langle \exists\ z\ :\ z = x :\ y = twice\ z \wedge even\ z \rangle \\
\equiv & \qquad \{ \text{ } \exists\text{-trading (8) ; introduce } \Phi_{even} \text{ } \} \\
& \langle \exists\ z\ ::\ y = twice\ z \wedge \underbrace{z = x \wedge even\ z}_{z(\Phi_{even})x} \rangle \\
\equiv & \qquad \{ \text{ composition (57) } \} \\
& y(twice \cdot \Phi_{even})x
\end{array}
$$

cf. diagram

$$
\begin{tikzcd}
\mathbb{N}_0 \arrow[d, "twice"'] & \mathbb{N}_0 \arrow[l, "\Phi_{even}"'] \\
\mathbb{N}_0 &
\end{tikzcd}
$$

Now the whole thing

$$
\begin{array}{ll}
& \langle \forall\ x, y\ :\ y = twice\ x \wedge even\ x :\ even\ y \rangle \\
\equiv & \qquad \{\ \text{above}\ \} \\
& \langle \forall\ x, y\ :\ y(twice \cdot \Phi_{even})x :\ even\ y \rangle \\
\equiv & \qquad \{\ \exists\text{-one-point}\ \} \\
& \langle \forall\ x, y\ :\ y(twice \cdot \Phi_{even})x :\ \langle \exists\ z\ :\ z = y :\ even\ z \rangle \rangle \\
\equiv & \qquad \{\ \text{predicate calculus: } p \wedge \mathrm{TRUE} = p\ \} \\
& \langle \forall\ x, y\ :\ y(twice \cdot \Phi_{even})x :\ \langle \exists\ z\ ::\ z = y \wedge even\ z \wedge \mathrm{TRUE} \rangle \rangle \\
\equiv & \qquad \{\ \top \text{ is the top relation}\ \} \\
& \langle \forall\ x, y\ :\ y(twice \cdot \Phi_{even})x :\ \langle \exists\ z\ ::\ y(\Phi_{even})z \wedge z \top x \rangle \rangle \\
\equiv & \qquad \{\ \text{composition}\ \}
\end{array}
$$

Now the whole thing

$$
\begin{array}{ll}
& \langle \forall\ x, y \ :\ y(twice \cdot \Phi_{even})x :\ y(\Phi_{even} \cdot \top)x \rangle \\
\equiv & \qquad \{ \text{ go pointfree (inclusion) } \} \\
& twice \cdot \Phi_{even} \ \subseteq\ \Phi_{even} \cdot \top
\end{array}
$$

cf. diagram

$$
\begin{tikzcd}
\mathbb{N}_0 \arrow[d, "twice"'] & \mathbb{N}_0 \arrow[l, "\Phi_{even}"'] \arrow[d, "\top"] \arrow[dl, phantom, "\subseteq"] \\
\mathbb{N}_0 & \mathbb{N}_0 \arrow[l, "\Phi_{even}"]
\end{tikzcd}
$$

In summary

In the calculation above, **unary** predicate *even* has been PF-transformed in two ways:

- Φ_{even} such that

$$z\ \Phi_{even} x \quad \triangleq \quad z = x \wedge even\ z$$

  Clearly, $\Phi_{even} \subseteq id$ — that is, Φ_{even} is a **coreflexive** relation;

- $\Phi_{even} \cdot \top$, which is such that

$$z(\Phi_{even} \cdot \top)x \ \equiv \ even\ z$$

  — a so-called (left) **condition**.

Coreflexives

As *id* can be represented as the "all-**1**s" diagonal matrix, so do **coreflexives**, which are *sub-diagonal* matrices, eg.

$\Phi_{vowel} =$

	a	b	c	d	e	f	...
a	1	0	0	0	0	0	0
b	0	0	0	0	0	0	0
c	0	0	0	0	0	0	0
d	0	0	0	0	0	0	0
e	0	0	0	0	1	0	0
f	0	0	0	0	0	0	0
...	0	0	0	0	0	0	...

where *vowel* is the predicate identifying characters which are vowels.

Coreflexives

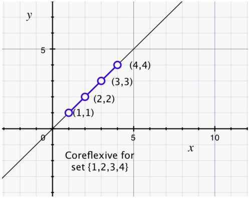
PF-transform of **unary** predicate p into the corresponding fragment Φ_p of id (coreflexive),

$$y\ \Phi_p\ x\ \equiv\ y = x \wedge p\ y \tag{144}$$

is unique — thus the universal property:

$$\Phi = \Phi_p\ \equiv\ (y\ \Phi\ x \equiv y = x \wedge p\ y) \tag{145}$$

A set S can also be PF-transformed into a coreflexive by calculating $\Phi_{(\in S)}$, cf. eg. the transform of set $\{1, 2, 3, 4\}$:

$\Phi_{1 \leq x \leq 4}\quad =$ 

Exercises

Exercise 58: Let $false$ be the "everywhere false" predicate such that $false\ x = \text{FALSE}$ for all x, that is, $false = \underline{\text{FALSE}}$. Show that $\Phi_{false} = \bot$.

□

Exercise 59: Given a set S, let Φ_S abbreviate coreflexive $\Phi_{(\in S)}$. Use (144) to unfold $\Phi_{\{1,2\}} \cdot \Phi_{\{2,3\}}$ to pointwise notation.

□

Exercise 60: Show that (145) follows from (144).

□

Exercise 61: Solve (145) for p under substitution $\Phi := id$.

□

Boolean algebra of coreflexives

Building up on the exercises above, from (145) one easily draws:

$$\Phi_{p \wedge q} = \Phi_p \cdot \Phi_q \tag{146}$$

$$\Phi_{p \vee q} = \Phi_p \cup \Phi_q \tag{147}$$

$$\Phi_{\neg p} = id - \Phi_p \tag{148}$$

$$\Phi_{false} = \bot \tag{149}$$

$$\Phi_{true} = id \tag{150}$$

where p, q are predicates.

(Note the slight, obvious abuse in notation.)

Basic properties of coreflexives

Let Φ, Ψ be coreflexive relations. Then the following properties hold:

- Coreflexives are **symmetric** and **transitive**:

$$\Phi^\circ = \Phi = \Phi \cdot \Phi \tag{151}$$

- **Meet** of two coreflexives is composition:

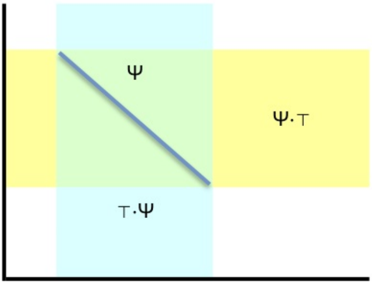
$$\Phi \cap \Psi = \Phi \cdot \Psi \tag{152}$$

- Closure properties:

$$R \cdot \Phi \subseteq S \quad \equiv \quad R \cdot \Phi \subseteq S \cdot \Phi \tag{153}$$

$$\Phi \cdot R \subseteq S \quad \equiv \quad \Phi \cdot R \subseteq \Phi \cdot S \tag{154}$$

Relating coreflexives with conditions



Coreflexive Ψ represented by a **right**-condition

$$\top \cdot \Psi$$

or by a left-condition:

$$\Psi \cdot \top$$

Mapping back and forward:

$$\Phi \subseteq \Psi \quad \equiv \quad \Phi \subseteq \top \cdot \Psi \qquad (155)$$

$$\Phi \subseteq \Psi \quad \equiv \quad \Phi \subseteq \Psi \cdot \top \qquad (156)$$

Relating coreflexives with conditions

Pre and post restriction:

$$R \cdot \Phi = R \cap \top \cdot \Phi \quad (157)$$

$$\Psi \cdot R = R \cap \Psi \cdot \top \quad (158)$$

Putting these together we obtain **selection**, as in SQL:

$$\sigma_{\Psi,\Phi} R \triangleq \Psi \cdot R \cdot \Phi \quad (159)$$

$$\begin{tikzcd} B \arrow[d, "\Psi"'] & A \arrow[l, "R"'] \arrow[d, "\Phi"] \\ B & A \arrow[l, "\sigma_{\Psi,\Phi} R"] \end{tikzcd}$$

Clearly,

$$\sigma_{\Psi,\Phi} R = \{(b,a) : b\ R\ a \wedge \psi\ b \wedge \phi\ a\} \quad (160)$$

for $\Psi = \Phi_\psi$ and $\Phi = \Phi_\phi$.

Selection

Let us check (160):

$$
\begin{array}{cl}
 & \sigma_{\Psi,\Phi} R \\
= & \qquad \{ \text{ set theoretical meaning of a relation } \} \\
 & \{(b,a) : b(\sigma_{\Psi,\Phi} R)a\} \\
= & \qquad \{ \text{ definition (159) } \} \\
 & \{(b,a) : b(\Psi \cdot R \cdot \Phi)a\} \\
= & \qquad \{ \text{ composition } \} \\
 & \{(b,a) : \langle \exists\ c\ :\ b\ \Psi\ c :\ c(R \cdot \Phi)a \rangle\} \\
= & \qquad \{ \text{ coreflexive } \Psi = \Phi_\psi \text{ (145) ; } \exists\text{-trading } \} \\
 & \{(b,a) : \langle \exists\ c\ :\ b = c :\ \psi b \wedge c(R \cdot \Phi)a \rangle\} \\
= & \qquad \{ \text{ next slide } \}
\end{array}
$$

Selection

$$
\begin{array}{cl}
= & \{\ \exists\text{-one-point ; composition again}\ \} \\
& \{(b,a) : \psi\ b \wedge \langle \exists\ d\ ::\ b\ R\ d \wedge d\ \Phi\ a \rangle\} \\
= & \{\ \text{coreflexive } \Phi = \Phi_\phi\ (145)\ ;\ \exists\text{-trading}\ \} \\
& \{(b,a) : \psi\ b \wedge \langle \exists\ d\ :\ d = a :\ b\ R\ d \wedge \phi\ a \rangle\} \\
= & \{\ \exists\text{-one-point ; trivia}\ \} \\
& \{(b,a) : \psi\ b \wedge b\ R\ a \wedge \phi\ a\}
\end{array}
$$

Exercise 62: Combinator

$$R \mathbin{\square} S \quad \triangleq \quad R \cdot \top \cdot S \tag{161}$$

is known as the "rectangular" combinator. Recalling that $\ker ! = \top$, show that $! \mathbin{\square} !^\circ \ = \ id$

□

Projection

By the way, another SQL-like relational operator is **projection**,

$$\pi_{g,f} R \triangleq g \cdot R \cdot f^{\circ} \qquad \begin{tikzcd} B \arrow[d, "g"'] & A \arrow[l, "R"'] \arrow[d, "f"] \\ C & D \arrow[l, "\pi_{g,f} R"] \end{tikzcd} \tag{162}$$

whose set-theoretic meaning is

$$\pi_{g,f} R = \{(g\ b, f\ a) : b\ R\ a\} \tag{163}$$

Functions f and g are often referred to as **attributes** of R.

Exercise 63: Derive (163) from (162).

□

Exercise

Exercise 64: A relation R is said to satisfy **functional dependency** (FD) $g \rightarrow f$, written $g \xrightarrow{R} f$ wherever projection $\pi_{f,g} R$ (162) is **simple**.

1. Show that

$$g \xrightarrow{R} f \quad \equiv \quad \ker(g \cdot R^{\circ}) \subseteq \ker f \tag{164}$$

holds.

2. Show that (164) trivially holds wherever g is injective and R is simple, for all (suitably typed) f.

3. Prove the **composition rule** of FDs:

$$h \xleftarrow{S \cdot R} g \quad \Leftarrow \quad h \xleftarrow{S} f \quad \wedge \quad f \xleftarrow{R} g \tag{165}$$

□

Two useful coreflexives

Domain:

$$\delta\, R \quad \triangleq \quad \ker R \cap id \tag{166}$$

Range:

$$\rho\, R \quad \triangleq \quad \mathrm{img}\, R \cap id \tag{167}$$

Universal properties:

$$\delta\, R \subseteq \Phi \quad \equiv \quad R \subseteq \top \cdot \Phi \tag{168}$$

$$\rho\, R \subseteq \Phi \quad \equiv \quad R \subseteq \Phi \cdot \top \tag{169}$$

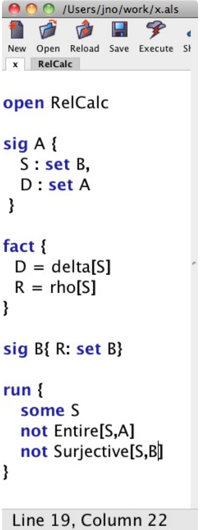
Domain/range elimination rules:

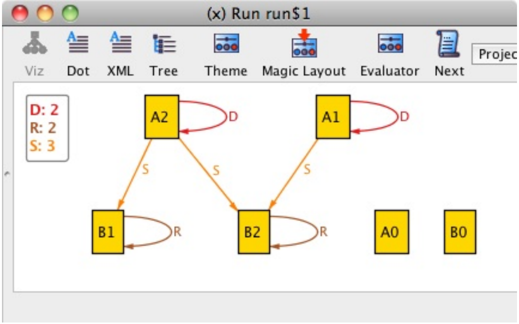
$$\top \cdot \delta\, R \quad = \quad \top \cdot R \tag{170}$$

$$\rho\, R \cdot \top \quad = \quad R \cdot \top \tag{171}$$

$$\delta\, R \subseteq \delta\, S \quad \equiv \quad R \subseteq \top \cdot S \tag{172}$$

$\delta\, R$ and $\rho\, R$ illustrated in Alloy





Two useful coreflexives

More facts about domain and range:

$$\delta\, R = \rho\,(R^\circ) \tag{173}$$

$$\delta\,(R \cdot S) = \delta\,(\delta\, R \cdot S) \tag{174}$$

$$\rho\,(R \cdot S) = \rho\,(R \cdot \rho\, S) \tag{175}$$

$$R = R \cdot (\delta\, R) \tag{176}$$

$$R = (\rho\, R) \cdot R \tag{177}$$

Exercise 65: Recalling (157), (158) and other properties of relation algebra, show that: (a) (168) and (169) can be re-written with R replacing $\top$; (b) $\Phi \subseteq \Psi \equiv ! \cdot \Phi \subseteq ! \cdot \Psi$ holds.

□

Exercise

Exercise 66: Recall diagram (117) of a library loan data model:

$$
\begin{tikzcd}
ISBN \arrow[d, "M"'] \arrow[dr, phantom, "\supseteq"] & ISBN \times UID \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] \arrow[d, "R"] \arrow[dr, phantom, "\subseteq"] & UID \arrow[d, "N"] \\
Title \times Publisher \arrow[r, "\top"'] & Date & Name \times Address \times Phone \arrow[l, "\top"]
\end{tikzcd}
$$

Show that the invariants captured by the two rectangles can be alternatively expressed by $\pi_{id,\pi_1} R \leq M$ and $\pi_{id,\pi_2} R \leq N$ where

$$R \leq S \quad \triangleq \quad \delta\, R \subseteq \delta\, S \tag{178}$$

clearly exhibiting the **foreign/primary**-key relationships of the data model ($ISBN$ and UID).

□

Coreflexives at work — data flow

Coreflexives are very handy in controlling information flow in PF-expressions, as the following two PF-transform rules show, given two suitably typed coreflexives $\Phi = \Phi_\phi$ and $\Psi = \Phi_\psi$:

- Guarded **composition**: for all b, c

$$\langle \exists\ a\ :\ \phi\ a :\ \ b\ R\ a \wedge a\ Sc \rangle \quad \equiv \quad b(R \cdot \Phi \cdot S)c \tag{179}$$

- Guarded **inclusion**:

$$\langle \forall\ b, a\ :\ \phi\ b \wedge \psi\ a :\ \ b\ R\ a \Rightarrow b\ S\ a \rangle \\ \equiv\ \Phi \cdot R \cdot \Psi \subseteq S \tag{180}$$

For $\Phi = id$ and $\Psi = id$ we recover the (non-guarded) standard definitions.

Coreflexives at work — satisfiability

Back to the **pre**/**post** specification style, by writing specification S

$$\begin{array}{l} S : (b : B) \leftarrow (a : A) \\ \textbf{pre} \ \ldots \\ \textbf{post} \ \ \ldots \end{array}$$

we mean the definition of two predicates

$$\begin{array}{l} \text{pre-}S : A \rightarrow \mathbb{B} \\ \text{post-}S : B \times A \rightarrow \mathbb{B} \end{array}$$

such that the **satisfiability** condition holds

$$\langle \forall\ a\ :\ a \in A \wedge \text{pre-}S\ a :\ \langle \exists\ b\ :\ b \in B :\ \text{post-}S(b, a) \rangle \rangle$$

recall (33).

Coreflexives at work — satisfiability

Let us abbreviate

- $\Phi_{\text{pre-}s}$ by Pre
- $\Phi_{\text{post-}s}$ by $Post$
- $\Phi_{(\in A)}$ by Φ_A, which in general includes an invariant associated to datatype A
- $\Phi_{(\in B)}$ by Φ_B, which in general includes an invariant associated to datatype B

Then (33) PF-transforms to

$$
\begin{tikzcd}
A \arrow[d, "Pre"'] & & A \arrow[ll, "\Phi_A"'] \arrow[d, "Post"] \\
A & B \arrow[l, "\top"] & B \arrow[l, "\Phi_B"]
\end{tikzcd}
\qquad
Pre \cdot \Phi_A \subseteq \top \cdot \Phi_B \cdot Post \tag{181}
$$

Functional satisfiability

Case $Pre = id$, $Post = f$:

$$
\begin{array}{cl}
 & \Phi_A \subseteq \top \cdot \Phi_B \cdot f \\
\equiv & \qquad \{ \text{ shunting rule (80) } \} \\
 & \Phi_A \cdot f^\circ \subseteq \top \cdot \Phi_B \\
\equiv & \qquad \{ \text{ converses } \} \\
 & f \cdot \Phi_A \subseteq \Phi_B \cdot \top \\
\equiv & \qquad \{ \text{ (96), since } f \cdot \Phi_A \subseteq f \text{ } \} \\
 & f \cdot \Phi_A \subseteq f \cap \Phi_B \cdot \top \\
\equiv & \qquad \{ \text{ (158) } \} \\
 & f \cdot \Phi_A \subseteq \Phi_B \cdot f
\end{array}
$$

What does this mean?

Functional satisfiability $\equiv$ invariant preservation

Let us introduce variables in $f \cdot \Phi_A \subseteq \Phi_B \cdot f$:

$$
\begin{array}{ll}
 & f \cdot \Phi_A \subseteq \Phi_B \cdot f \\
\equiv & \quad \{ \text{ shunting rule (79) } \} \\
 & \Phi_A \subseteq f^\circ \cdot \Phi_B \cdot f \\
\equiv & \quad \{ \text{ introduce variables } \} \\
 & \langle \forall\ a, a' \ :\ a\ \Phi_A\ a' :\ (f\ a) \Phi_B (f\ a') \rangle \\
\equiv & \quad \{ \text{ coreflexives } (a = a') \ \} \\
 & \langle \forall\ a\ ::\ a\ \Phi_A\ a \Rightarrow (f\ a) \Phi_B (f\ a) \rangle \\
\equiv & \quad \{ \text{ meaning of } \Phi_A,\ \Phi_B \ \} \\
 & \langle \forall\ a\ :\ a \in A :\ (f\ a) \in B \rangle
\end{array}
$$

Functional satisfiability $\equiv$ invariant preservation

Another way to put it:

$$
\begin{array}{ll}
& f \cdot \Phi_A \subseteq \Phi_B \cdot f \\
\equiv & \qquad \{ \text{ shunting } \} \\
& f \cdot \Phi_A \cdot f^\circ \subseteq \Phi_B \\
\equiv & \qquad \{ \text{ coreflexives } \} \\
& f \cdot \Phi_A \cdot \Phi_A^\circ \cdot f^\circ \subseteq \Phi_B \\
\equiv & \qquad \{ \text{ image definition } \} \\
& \mathrm{img}\,(f \cdot \Phi_A) \subseteq \Phi_B \\
\equiv & \qquad \{ \; f \cdot \Phi_A \text{ is simple } \} \\
& \rho\,(f \cdot \Phi_A) \subseteq \Phi_B
\end{array}
$$

Functional satisfiability $\equiv$ invariant preservation

We will write "type declaration"

$$\Phi_B \xleftarrow{f} \Phi_A \tag{182}$$

to mean

$$f \cdot \Phi_A \subseteq \Phi_B \cdot f \quad \text{cf. diagram} \quad \begin{tikzcd} A \arrow[d, "f"'] & A \arrow[l, "\Phi_A"'] \arrow[d, "f"] \\ B & B \arrow[l, "\Phi_B"] \end{tikzcd} \tag{183}$$

equivalent to both

$$f \cdot \Phi_A \subseteq \Phi_B \cdot \top \tag{184}$$

$$\rho\,(f \cdot \Phi_A) \subseteq \Phi_B \tag{185}$$

Design by contract

In general, a "type declaration" $\Psi \xleftarrow{f} \Phi$ (182) is the basis of **functional programming** (f) with so-called **contracts** (Ψ, Φ), an instance of the well-known *Design by Contract* (**DbC**) methodology (more about this later).

DbC works because **contracts** are compositional,

$$\Psi \xleftarrow{f \cdot g} \Phi \quad \Leftarrow \quad \Psi \xleftarrow{f} \Upsilon \wedge \Upsilon \xleftarrow{g} \Phi \tag{186}$$

that is, diagram

$$\begin{tikzcd} \Psi & \Upsilon \arrow[l, "f"'] & \Phi \arrow[l, "g"'] \arrow[ll, bend left, "f \cdot g"] \end{tikzcd}$$

makes sense.

Design by contract

Contract composition (186) is easy to prove:

$$
\begin{array}{ll}
 & \Psi \xleftarrow{\ f\ } \Upsilon \ \wedge\ \Upsilon \xleftarrow{\ g\ } \Phi \\
\equiv & \qquad \{\ (182) \text{ twice } \} \\
 & f \cdot \Upsilon \subseteq \Psi \cdot f \ \wedge\ g \cdot \Phi \subseteq \Upsilon \cdot g \\
\Rightarrow & \qquad \{\ \text{monotonicity of } (\cdot g) \text{ and } (f \cdot)\ \} \\
 & f \cdot \Upsilon \cdot g \subseteq \Psi \cdot f \cdot g \ \wedge\ f \cdot g \cdot \Phi \subseteq f \cdot \Upsilon \cdot g \\
\Rightarrow & \qquad \{\ \subseteq \text{ is transitive } \} \\
 & f \cdot g \cdot \Phi \subseteq \Psi \cdot f \cdot g \\
\equiv & \qquad \{\ (182)\ \} \\
 & \Psi \xleftarrow{\ f \cdot g\ } \Phi
\end{array}
$$

Design by contract

Contracts cam also be paired, leading to the type rule (188) which is derived in the exercise below.

Exercise 67: Rely on the **absorption** property

$$\langle R \cdot T, S \cdot U \rangle = (R \times S) \cdot \langle T, U \rangle \tag{187}$$

in showing that

$$\Psi \times \Upsilon \xleftarrow{\langle f, g \rangle} \Phi \equiv \Psi \xleftarrow{f} \Phi \wedge \Upsilon \xleftarrow{g} \Phi \tag{188}$$

holds.

□

Exercises

Exercise 68: From (182) and properties (79), etc infer the following **DbC** rules

$$\Upsilon \xleftarrow{f} \Phi \cup \Psi \quad \equiv \quad \Upsilon \xleftarrow{f} \Phi \ \wedge \ \Upsilon \xleftarrow{f} \Psi \tag{189}$$

$$\Phi \cdot \Psi \xleftarrow{f} \Upsilon \quad \equiv \quad \Phi \xleftarrow{f} \Upsilon \ \wedge \ \Psi \xleftarrow{f} \Upsilon \tag{190}$$

You will also need $(R\cdot)$-distribution (101).

□

Exercise 69: Show that (181) means the same as

$$Pre \cdot \Phi_A \subseteq Post^\circ \cdot \Phi_B \cdot Post \tag{191}$$

□

Exercises

Exercise 70: Consider the relational version of McCarthy's conditional combinator which follows:

$$p \rightarrow f, g \;=\; f \cdot \Phi_p \cup g \cdot \Phi_{\neg p} \tag{192}$$

(a) Using (184) infer the following **DbC** rule for *conditionals*:

$$\Upsilon \xleftarrow{p \rightarrow f, g} \Psi \quad \equiv \quad \Upsilon \xleftarrow{f} \Psi \cdot \Phi_p \;\wedge\; \Upsilon \xleftarrow{g} \Psi \cdot \Phi_{\neg p} \tag{193}$$

(b) Now try and define a rule for handling contracts involving conditional conditions:

$$\Upsilon \xleftarrow{p \rightarrow f, g} (p \rightarrow \Psi, \Phi) \;=\; \ldots. \tag{194}$$

□

Exercises

Exercise 71: Recall that our motivating ESC assertion (141) was stated but not proved. Now that we know that (141) PF-transforms to

$\Phi_{even} \xleftarrow{twice} \Phi_{even}$ and that $\Phi_{even} = \rho\, twice$, complete the following *"almost no work at all"* PF-calculation of (141):

$$
\begin{array}{cl}
 & \Phi_{even} \xleftarrow{twice} \Phi_{even} \\
\equiv & \{ \ \ldots\ldots\ldots \ \} \\
 & twice \cdot \Phi_{even} \subseteq \Phi_{even} \cdot twice \\
\equiv & \{ \ \ldots\ldots\ldots \ \} \\
 & twice \cdot \Phi_{even} \subseteq \rho\, twice \cdot twice
\end{array}
\qquad
\begin{array}{cl}
\equiv & \{ \ \ldots\ldots\ldots \ \} \\
 & twice \cdot \Phi_{even} \subseteq twice \\
\Leftarrow & \{ \ \ldots\ldots\ldots \ \} \\
 & \Phi_{even} \subseteq id \\
\equiv & \{ \ \ldots\ldots\ldots \ \} \\
 & \textsc{True}
\end{array}
$$

□