4.10. Duality Revisited: Conditionals, Biconditionals, and More

1. Duals of Conditionals. Among the many remarkable features of the Chapter Three language is the fact that each connective finds its dual connective within that language. (In technical parlance: the Chapter Three language is "*closed* under duality" using the Connective Swap Method.) But that is not so for the Chapter Four language, which adds conditionals and biconditionals.

Using the True/False Swap Method on the truth table for "(P → Q)" yields a dual truth table which is true in only one valuation: where "Q" is true and "P" is false.

P	Q	(P → Q)
1	1	1
1	0	0
0	1	1
0	0	1

P	Q	(P → Q)
False	False	False
False	**True**	**True**
True	False	False
True	True	False

Copyright Brian Beakley 2015

Versed as we are in methods for matching truth tables with sentences, we know that a truth table with only one true valuation is matched by a **valuation sentence** – in this case, "(Q ∧ ~P)".[1]

P	Q	(P → Q)	~P	(Q ∧ ~P)
1	1	**1**	0	**0**
1	0	**0**	0	**0**
0	1	**1**	1	**1**
0	0	**1**	1	**0**

[1] As discussed in *3.26. Valuation and Anti-Valuation Sentences*.

However, we cannot here move to the Connective Swap Method for stating the dual of “(P → Q),” since we have no *single* connective that takes the truth table for “(Q ∧ ~P)”.

But in keeping with the greater freedom we have lately felt to introduce new connectives (e.g., bicon), let us now expand the Chapter Four language to include a single connective taking this truth table. Now, “(Q ∧ ~P)” is the formal translation of a “**without**” sentence such as “Rex passed Chemistry without studying” or “Neko swallowed her food without chewing it”. So as the formal counterpart to “without” we introduce the connective “**%**” – whose resemblance to the abbreviation “**w/o**” should remind us of its meaning. To further stress the connection between “**%**” and “w/o,” the “**%**” symbol will be called “**wo**”.[2]

●	▲	(● % ▲)
1	1	0
1	0	1
0	1	0
0	0	0

Note that wo sentences, like conditionals, are sensitive to the order of their parts: just as “(P → Q)” and “(Q → P)” take different truth tables, so likewise do “(P **%** Q)” and “(Q **%** P)”.

P	Q	(P % Q)	(Q % P)
1	1	**0**	**0**
1	0	**1**	**0**
0	1	**0**	**1**
0	0	**0**	**0**

That makes sense semantically: the English sentences “Rex passed Chemistry without studying” and “Rex studied without passing Chemistry” certainly don’t have the same meaning.

And the fact that order here makes a difference affects logical duality. For recall that the dual of “(P → Q)” was not “(P **%** Q),” but “(Q **%** P)”. **When taking the dual of a conditional via Connective Swap, the two parts of the conditional must switch places**.

[2] The wo is sometimes referred to as “difference” (or “asymmetric difference”).

2. Duals of Biconditionals. The True/False Swap Method for "(P ↔ Q)" likewise yields a truth table had by no *single* connective in our current formal language. This is the truth table for a sentence only true when one or the other of its parts is true – but not when both are.

P	Q	(P ↔ Q)
1	1	1
1	0	0
0	1	0
0	0	1

P	Q	(P ↔ Q)
False	False	False
False	**True**	**True**
True	**False**	**True**
True	True	False

Our phrasing of those truth conditions calls to mind the **exclusive "or"**: "P or Q, but not both". And exclusive "or" does indeed fit the bill.

P	Q	(P ↔ Q)	(P ∨ Q)	(P ∧ Q)	~(P ∧ Q)	((P ∨ Q) ∧ ~(P ∧ Q))
1	1	**1**	1	1	0	**0**
1	0	**0**	1	0	1	**1**
0	1	**0**	1	0	1	**1**
0	0	**1**	1	0	1	**0**

Since our formal language lacks a single connective for such a sentence, we introduce the symbol "⊕" – called "**exor**" – to express an exclusive disjunction.[3] The exor will serve as the dual of the bicon, in the Connective Swap Method.

●	▲	(● ⊕ ▲)
1	1	0
1	0	1
0	1	1
0	0	0

[3] This is sometimes called "symmetric difference".

3. Further Connectives. In both of the above cases of duality, we already had a formal sentence taking the dual truth table in question – indeed, infinitely many such sentences. We introduced special connectives *only* so we could have a single-connective sentence associated with that truth table, in order to pair the arrow and bicon with another connective for purposes of Connective Swap.

But once we've set our sights on matching each truth table with a single-connective sentence, we spot certain truth tables which are not yet so matched. We list here all the four-valuation truth tables.

?	?	$(P \rightarrow Q)$	$(Q \rightarrow P)$	$(P \vee Q)$	$(P \leftrightarrow Q)$	$\sim P$	P	Q	$\sim Q$	$(P \oplus Q)$	$(P \wedge Q)$	$(P \% Q)$	$(Q \% P)$	?	?
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
1	0	1	1	1	1	0	1	1	0	0	1	0	0	0	0
1	1	0	1	0	0	0	1	0	1	1	0	1	0	0	0
1	1	1	0	0	0	1	0	1	0	1	0	0	1	0	0
1	1	1	1	0	1	1	0	0	1	0	0	0	0	1	0

Copyright Brian Beakley 2015

Truth Tables 1, 2, 15, and 16 are not yet associated with a single connective.

It is true that if we restrict ourselves to one sentence letter, "$(P \rightarrow P)$" will take Truth Table 1 and "(P % P)" will take 16. But in the interest of assigning a unique connective to each truth table, we add a new connective in each of the four remaining cases.

Truth Table 1 is true *regardless* of what "P" and "Q" are; and likewise for the falsehood in Truth Table 16. So a single connective taking Truth Table 1 needn't look at sentence letters at all; and likewise for a single connective taking Truth Table 16. Whereas a tilde is a **one**-place connective (yielding a complete sentence with its own truth table when one sentence is added to the tilde), and all of the other connectives are **two**-placed (two sentences short of a complete sentence), Tables 1 and 16 will each take a **logical constant** – a **zero-placed connective** with no blanks, hence needing no sentences added to take a truth table.

To Truth Table 1 we assign the logical constant "⊤" (pronounced "**tee**"). "⊤" is always true, regardless of what value "P" or "Q" take.

●	▲	⊤
1	1	1
1	0	1
0	1	1
0	0	1

To Truth Table 16 we assign the logical constant "⊥" (which – since it is an inverted "⊤" – will take the inverted name "**eet**").[4]

●	▲	⊥
1	1	0
1	0	0
0	1	0
0	0	0

Truth Table 15 takes the "neither... nor" sentence "~(P ∨ Q)".

P	Q	**~(P ∨ Q)**
1	1	0
1	0	0
0	1	0
0	0	1

[4] Following Smullyan 1992 [Goedel's Incompleteness Theorems]: 132.

For this truth table we introduce the two-placed connective "↓," called the "**dagger**"[5]. (The name "dagger" suggests an easy mnemonic for recalling its truth conditions: the **d**ownward **d**agger **d**enies a **d**isjunction.)

●	▲	**(P ↓ Q)**
1	1	0
1	0	0
0	1	0
0	0	1

Finally, Truth Table 2 takes the "not both" sentence "~(P ∧ Q)".

P	Q	**~(P ∧ Q)**
1	1	0
1	0	1
0	1	1
0	0	1

For this truth table we add a two-placed connective "|" called the "**stroke**".[6]

●	▲	**(P | Q)**
1	1	0
1	0	1
0	1	1
0	0	1

[5] This is sometimes called a "**NOR**".
[6] This is sometimes called a "**NAND**" (on analogy with "NOR").

We now have a single (zero- or one- or two-placed) connective assigned to each possible (four-valuation) truth table.[7]

⊤	(P | Q)	(P → Q)	(Q → P)	(P ∨ Q)	(P ↔ Q)	~P	P	Q	~Q	(P ⊕ Q)	(P ∧ Q)	(P % Q)	(Q % P)	(P ↓ Q)	⊥
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
1	0	1	1	1	1	0	1	1	0	0	1	0	0	0	0
1	1	0	1	0	0	0	1	0	1	1	0	1	0	0	0
1	1	1	0	0	0	1	0	1	0	1	0	0	1	0	0
1	1	1	1	0	1	1	0	0	1	0	0	0	0	1	0

In terms of duality, the True/False Swap Method makes clear that **the dual of** "⊤" **is** "⊥" (and vice versa) – for a truth table with "1" in every valuation becomes a truth table with "0" in every valuation (and vice versa).

Likewise **the dagger is the dual of the stroke**. For applying the True/False swap to the truth table for "(P ↓ Q)" yields a sentence only false when both its parts are true – the "*not both*" truth table of "(P | Q)".

P	Q	(P ↓ Q)
1	1	0
1	0	0
0	1	0
0	0	1

P	Q	(P ↓ Q)
False	False	True
False	True	True
True	False	True
True	**True**	**False**

We are therefore equipped with a master set of connectives – call it "**Formal Language A**" – where (i) every (four-valuation) truth table has a single-connective sentence, and where (ii) every connective finds its dual within that language.

A: { |, →, ↔, ∨, ⊤, ~, ⊥, ∧, ⊕, %, ↓}

[7] If we liked we could replaced the entries for "P" and "Q" – which contain no connectives – by introducing a one-placed connective "**i**" such that "**i**●" is logically equivalent to "●". "P" could then be replaced by "**i**P," and "Q" by "**i**Q," so that every entry contained a connective. The connective "**i**" is its own dual.

Language A is therefore closed under duality, using the Connective Swap Method.

We turn next to the effect these new connectives have on issues of **expressive adequacy**.