Program transformation is not compositional. Assume program P to be a trivial, terminating program.
Assume that program Q is:

```
1. Observe (x == 0)
2. if (x) {
3.     diverge;
4. }
5. x := 1;
```

When considering $wp(Q, f)$, as should be done in the case $wp(\overbrace{P;Q}, f)$, the wp will consider the diverge-statement on line 3.
However, when considering $wp(\hat{Q}, f)$, the diverge-statement will not be considered as the if-statement can safely be ignored. This can be seen when constructing the weakest pre-expectations:

$$wp(Q, f) = wp(\text{if } x \text{ then diverge else skip; end}, f)$$
$$= [x] \cdot wp(\text{diverge}, f) + [\neg x] \cdot wp(\text{skip}, f)$$
$$= [x] \cdot 0 + [\neg x] \cdot f = [\neg x] \cdot f.$$

$$wp(\hat{Q}, f) = wp(\text{if } x \text{ then diverge else skip; } x := 0; \text{ end}, f)$$
$$= wp(\text{if } x \text{ then diverge else skip; }, f[x := 0])$$
$$= [x] \cdot wp(\text{diverge}, f[x := 0]) + [\neg x] \cdot wp(\text{skip}, f[x := 0])$$
$$= 0 \cdot 0 + 1 \cdot f[x := 0] = f[x := 0]$$

Therefore we can conclude that the moment of transformation is significant and not compositional, therefore:

$$wp(\overbrace{P;Q}, f) \neq wp(\hat{P};\hat{Q}, f)$$