This works because now when evaluating `(magnitude z)`, where z is a complex number with outer tag `complex` and inner tag `rectangular`, there is now an operation in the dispatch table for `magnitude` for type `(complex)`, which results in applying the generic procedure `magnitude` (which operates on both rectangular and polar representations of complex numbers) on z. This in turn requires another `apply-generic` call, which will dispatch `magnitude-rectangular` and return 5. Here is the trace of all the procedures for `(magnitude z)`. There is only one argument in each `apply-generic` call, so the `map` calls are omitted.

```
(magnitude z)
(apply-generic 'magnitude z)
((lambda (proc) (apply proc ('rectangular (cons 3 4)))) (get 'magnitude 'complex))
(apply magnitude ('rectangular (cons 3 4)))
(apply-generic 'magnitude ('rectangular (cons 3 4)))
((lambda (proc) (apply proc (cons 3 4))) (get 'magnitude 'rectangular))
(apply 'magnitude (cons 3 4))
(sqrt (+ (square (real-part (cons 3 4)))
         (square (imag-part (cons 3 4)))))
(sqrt (+ (square (car (cons 3 4)))
         (square (cdr (cons 3 4)))))
(sqrt (+ 9 16))
```
5

Essentially, `magnitude` is applied twice, first to strip away the `complex` tag, and again after stripping away the `rectangular` tag and dispatching the correct `magnitude` procedure (for the given representation).