```
ComposeMaps (M1, M2, D, T, R):  trait
  % If m1 is a map from D to T
  % and m2 is a map from T to R,
  % m1 ∘ m2 is a map from D to R.
  assumes
    FiniteMap (M1, T, R),
    FiniteMap (M2, D, T)
  includes FiniteMap
  introduces __ ∘ __: M1, M2 → M
  asserts ∀ m1: M1, m2: M2, d: D
    apply(m1 ∘ m2, d) == apply(m1, apply(m2, d));
    defined(m1 ∘ m2, d) ==
      defined(m2, d) ∧ defined(m1, apply(m2, d))
```