```
FiniteMap (M, D, R): trait
  % An M is a map from D's to R's.
  introduces
    { }: → M
    update: M, D, R → M
    apply: M, D → R
    defined: M, D → Bool
  asserts
    M generated by { }, update
    M partitioned by apply, defined
    ∀ m: M, d, d1, d2: D, r: R
      apply(update(m, d2, r), d1) ==
        if d1 = d2 then r else apply(m, d1);
      ¬defined({ }, d);
      defined(update(m, d2, r), d1) ==
        d1 = d2 ∨ defined(m, d1)
  implies
    Array1 (update for assign, apply for __[__],
            M for A, D for I, R for E)
    converts apply, defined
      exempting ∀ d: D apply({ }, d)
```