```
fpf-join-cap-sq  THM

⊢  ∀A:Type, eq:EqDecider(A), f,g:a:A fp-> Top, x:A, z:Top.
   f ⊕ g(x)?z ∼ if x ∈ dom(f)→ f(x)?z else g(x)?z fi
  by UnivCD THEN Unfold ‘fpf-cap‘ 0
     THEN
     RWO
     Thm* ∀eq:EqDecider(A), f:a:A fp-> Top, g:Top, x:A.
          f ⊕ g(x) ∼ if x ∈ dom(f)→ f(x) else g(x) fi
     {fpf_DASH_join_DASH_ap_DASH_sq}
     0
     THEN
     Repeat SplitOnConclITE
     THEN
     Try Trivial
     THEN
     All
     (RWO
      Thm* ∀B:(A→Type), eq:EqDecider(A), f,g:a:A fp-> B(a), x:A.
           x ∈ dom(f ⊕ g) ⟺ x ∈ dom(f) ∨ x ∈ dom(g)
      {fpf_DASH_join_DASH_dom})
     THEN
     SplitOrHyps
     THEN
     Try Trivial
     THEN
     AllHyps (λh.Analyze h THEN OrLeft THEN Complete Auto)
     THEN
     AllHyps (λh.Analyze h THEN OrRight THEN Complete Auto)
END
```