```
fpf-join-ap  THM

⊢  ∀A:Type, B:(A→Type), eq:EqDecider(A), f,g:a:A fp-> B(a), x:A.
|  x ∈ dom(f ⊕ g) ⇒ f ⊕ g(x) = if x ∈ dom(f)→ f(x) else g(x) fi
| by Auto THEN Unfolds [‘fpf-join‘;‘fpf-ap‘] 0 THEN Reduce 0
|    THEN
|    Subst ((2of(f)(x)) ∼ f(x)) 0
|    THEN
|    Try (Unfold ‘fpf-ap‘ 0 THEN Trivial)
|    THEN
|    Subst ((2of(g)(x)) ∼ g(x)) 0
|    THEN
|    Try (Unfold ‘fpf-ap‘ 0 THEN Trivial)
|    THEN
|    Unfold ‘fpf-cap‘ 0
|    THEN
|    Fold ‘member‘ 0
|    THEN
|    SplitOnConclITE
|
1. A : Type
2. B : A→Type
3. eq : EqDecider(A)
4. f : a:A fp-> B(a)
5. g : a:A fp-> B(a)
6. x : A
7. x ∈ dom(f ⊕ g)
8. ¬x ∈ dom(f)
⊢  x ∈ dom(g)
  by (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}
      -2)
     THEN
     (Analyze -2)
END
```