```
fpf-compatible-triple  THM

⊢  ∀T:Type, eq:EqDecider(T), f,g,h:x:T fp-> Type.
|  f || g
|  ⇒
|  h || f
|  ⇒
|  h || g
|  ⇒
|  g ⊆ h ⊕ f ⊕ g & f ⊆ h ⊕ f ⊕ g
|  & h ⊕ g ⊆ h ⊕ f ⊕ g
|  & h ⊕ f ⊆ h ⊕ f ⊕ g  by Auto
|
1. T : Type
2. eq : EqDecider(T)
3. f : x:T fp-> Type
4. g : x:T fp-> Type
5. h : x:T fp-> Type
6. f || g
7. h || f
8. h || g
⊢  g ⊆ h ⊕ f ⊕ g & f ⊆ h ⊕ f ⊕ g
   & h ⊕ g ⊆ h ⊕ f ⊕ g
   & h ⊕ f ⊆ h ⊕ f ⊕ g
  by Unfolds [‘guard‘;‘fpf-sub‘] 0
     THEN
     Try
     (RWW
      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 Symmetry
     THEN
     AllHyps (λh.Unfold ‘fpf-compatible‘ h THEN BackThru h)
     THEN
     All
     (RWW
      Thm* ∀eq:EqDecider(A), f,g:a:A fp-> Top, x:A.
           x ∈ dom(f ⊕ g) ⟺ x ∈ dom(f) ∨ x ∈ dom(g)
      {fpf_DASH_join_DASH_dom2})
     THEN
     SplitOrHyps
     THEN
     Try (Sel 1 (Analyze 0) THEN Complete Auto)
     THEN
     Try (Sel 2 (Analyze 0) THEN Complete Auto)
     THEN
     Try (Sel 3 (Analyze 0) THEN Complete Auto)
END
```