```
w-causl-time  THM

⊢  ∀the_w:World, e,e':E. FairFifo ⇒ e <c e' ⇒ time(e)<time(e')
| by Auto
|
1. the_w : World
2. e : E
3. e' : E
4. FairFifo
5. e <c e'
⊢  time(e)<time(e')
| by Unfold `w-causl` -1 THEN Unfold `rel_plus` -1 THEN Reduce -1
|    THEN
|    Assert
|    (∀n:ℕ+, e,e':E.
|     (e λe,e'. e <loc e' ∨ isrcv(kind(e')) & e = sender(e')^n e')
|     ⇒
|     time(e)<time(e'))
|
|\
| .....assertion .....
| 1. the_w : World
| 2. e : E
| 3. e' : E
| 4. FairFifo
| 5. ∃n:ℕ+. e λe,e'. e <loc e' ∨ isrcv(kind(e')) & e = sender(e')^n e'
| ⊢  ∀n:ℕ+, e,e':E.
| |  (e λe,e'. e <loc e' ∨ isrcv(kind(e')) & e = sender(e')^n e')
| |  ⇒
| |  time(e)<time(e')
| | by Thin -1 THEN Thin -2 THEN Thin -2
| |    THEN
| |    Assert
| |    (∀e,z:E.
| |     e <loc z ∨ isrcv(kind(z)) & e = sender(z) ⇒ time(e)<time(z))
| |
| |\
| | .....assertion .....
| | 1. the_w : World
| | 2. FairFifo
| | ⊢  ∀e,z:E.
| | |  e <loc z ∨ isrcv(kind(z)) & e = sender(z) ⇒ time(e)<time(z)
| | | by Auto THEN Analyze -1 THEN ExRepD
| | |
| | |\
| | | 1. the_w : World
| | | 2. FairFifo
| | | 3. e : E
| | | 4. z : E
| | | 5. e <loc z
| | | ⊢  time(e)<time(z)  by Unfold `w-locl` -1
| | |
| | .\
| | . 1. the_w : World
| | . 2. FairFifo
```

```
| | . 3. e : E
| | . 4. z : E
| | . 5. isrcv(kind(z))
| | . 6. e = sender(z)
| | . ⊢  time(e)<time(z)  by RenameTo ‘e\’’ ‘z’ THEN HypSubst -1 0
| | . |
| | . 1. the_w : World
| | . 2. FairFifo
| | . 3. e : E
| | . 4. e’ : E
| | . 5. isrcv(kind(e’))
| | . 6. e = sender(e’)
| | . ⊢  time(sender(e’))<time(e’)
| | . | by AssertBY (∃n:ℕtime(e’). match(lnk(kind(e’));n;time(e’)))
| | . |    (BackThru
| | . |     Thm* ∀the_w:World, e:E.
| | . |           FairFifo
| | . |           ⇒
| | . |           isrcv(kind(e))
| | . |           ⇒
| | . |           (∃t:ℕtime(e). match(lnk(kind(e));t;time(e)))
| | . |     {w_DASH_match_DASH_exists})
| | . |    THEN
| | . |    Inst
| | . |    Thm* ∀f:(ℕ→𝔹).
| | . |          (∃n:ℕ. f(n)) ⇒ f(mu(f)) & (∀i:ℕ. i<mu(f) ⇒ ¬f(i))
| | . |    {mu_DASH_property}
| | . |    [λt.match(lnk(kind(e’));t;time(e’))]
| | . |    THENA
| | . |    (Reduce 0 THEN ParallelOp -1)
| | . |    THEN
| | . |    Reduce -1
| | . |    THEN
| | . |    ExRepD
| | . |
| | . 1. the_w : World
| | . 2. FairFifo
| | . 3. e : E
| | . 4. e’ : E
| | . 5. isrcv(kind(e’))
| | . 6. e = sender(e’)
| | . 7. n : ℕtime(e’)
| | . 8. match(lnk(kind(e’));n;time(e’))
| | . 9. match(lnk(kind(e’));mu(λt.
| | . |  match(lnk(kind(e’));t;time(e’)));time(e’))
| | . 10. ∀i:ℕ.
| | . |    i<mu(λt.match(lnk(kind(e’));t;time(e’)))
| | . |    ⇒
| | . |    ¬match(lnk(kind(e’));i;time(e’))
| | . ⊢  time(sender(e’))<time(e’)  by Assert (time(sender(e’))≤n)
| | . |
| | . .....assertion .....
| | . 1. the_w : World
| | . 2. FairFifo
| | . 3. e : E
```

```
| | . 4. e' : E
| | . 5. isrcv(kind(e'))
| | . 6. e = sender(e')
| | . 7. n : ℕtime(e')
| | . 8. match(lnk(kind(e'));n;time(e'))
| | . 9. match(lnk(kind(e'));mu(λt.
| | . |  match(lnk(kind(e'));t;time(e')));time(e'))
| | . 10. ∀i:ℕ.
| | . |    i<mu(λt.match(lnk(kind(e'));t;time(e')))
| | . |    ⇒
| | . |    ¬match(lnk(kind(e'));i;time(e'))
| | . ⊢  time(sender(e'))≤n  by SupposeNot THEN InstHyp [n] -2
| | . |
| | . .....antecedent .....
| | . 1. the_w : World
| | . 2. FairFifo
| | . 3. e : E
| | . 4. e' : E
| | . 5. isrcv(kind(e'))
| | . 6. e = sender(e')
| | . 7. n : ℕtime(e')
| | . 8. match(lnk(kind(e'));n;time(e'))
| | . 9. match(lnk(kind(e'));mu(λt.
| | . |  match(lnk(kind(e'));t;time(e')));time(e'))
| | . 10. ∀i:ℕ.
| | . |    i<mu(λt.match(lnk(kind(e'));t;time(e')))
| | . |    ⇒
| | . |    ¬match(lnk(kind(e'));i;time(e'))
| | . 11. ¬time(sender(e'))≤n
| | . ⊢  n<mu(λt.match(lnk(kind(e'));t;time(e')))
| | . | by Unfolds [`w-sender`;`w-time`] -1 THEN Reduce -1
| | . |
| | . 1. the_w : World
| | . 2. FairFifo
| | . 3. e : E
| | . 4. e' : E
| | . 5. isrcv(kind(e'))
| | . 6. e = sender(e')
| | . 7. n : ℕtime(e')
| | . 8. match(lnk(kind(e'));n;time(e'))
| | . 9. match(lnk(kind(e'));mu(λt.
| | .    match(lnk(kind(e'));t;time(e')));time(e'))
| | . 10. ∀i:ℕ.
| | .      i<mu(λt.match(lnk(kind(e'));t;time(e')))
| | .      ⇒
| | .      ¬match(lnk(kind(e'));i;time(e'))
| | . 11. ¬mu(λt.match(lnk(kind(e'));t;time(e')))≤n
| | . ⊢  ∃n:ℕ. (λt.match(lnk(kind(e'));t;time(e')))(n)
| | .    by Reduce 0 THEN InstConcl [n]
| |
| .\
| . 1. the_w : World
| . 2. FairFifo
| . 3. ∀e,z:E.
| . |  e <loc z ∨ isrcv(kind(z)) & e = sender(z) ⇒ time(e)<time(z)
```

```
| . ⊢  ∀n:ℕ+, e,e':E.
| . |  (e λe,e'. e <loc e' ∨ isrcv(kind(e')) & e = sender(e')^n e')
| . |  ⇒
| . |  time(e)<time(e')
| . | by InductionOnNat THEN RecUnfold `rel_exp` 0 THEN Reduce 0
| . |
| . |\
| . | 1. the_w : World
| . | 2. FairFifo
| . | 3. ∀e,z:E.
| . |    e <loc z ∨ isrcv(kind(z)) & e = sender(z) ⇒ time(e)<time(z)
| . | 4. n : ℤ
| . | 5. 0<n
| . | 6. 0<n-1
| . |    ⇒
| . |    (∀e,e':E.
| . |     (e
| . |      λe,e'. e <loc e' ∨ isrcv(kind(e')) & e = sender(e')^n-1 e')
| . |     ⇒
| . |     time(e)<time(e'))
| . | 7. n = 1
| . | ⊢  0<1
| . |    ⇒
| . |    (∀e,e':E.
| . |     (∃z:E.
| . |      e <loc z ∨ isrcv(kind(z)) & e = sender(z)
| . |      & (z
| . |         λe,e'. e <loc e'
| . |         ∨ isrcv(kind(e')) & e = sender(e')^0 e'))
| . |     ⇒
| . |     time(e)<time(e'))
| . |   by Auto THEN ExRepD THEN RecUnfold `rel_exp` -1 THEN Reduce -1
| . |      THEN
| . |      RevHypSubst -1 0
| . |      THEN
| . |      BackThru 3
| . |

| . .\
| . . 1. the_w : World
| . . 2. FairFifo
| . . 3. ∀e,z:E.
| . . |  e <loc z ∨ isrcv(kind(z)) & e = sender(z) ⇒ time(e)<time(z)
| . . 4. n : ℤ
| . . 5. 0<n
| . . 6. 0<n-1
| . . |  ⇒
| . . |  (∀e,e':E.
| . . |   (e
| . . |    λe,e'. e <loc e' ∨ isrcv(kind(e')) & e = sender(e')^n-1 e')
| . . |   ⇒
| . . |   time(e)<time(e'))
| . . 7. ¬n = 1
| . . ⊢  0<n
| . . |  ⇒
| . . |  (∀e,e':E.
```

```
| . . |   (e
| . . |    if n=_2 0→ λx,y. x = y
| . . |    else λx,y. ∃z:E.
| . . |         x <loc z ∨ isrcv(kind(z)) & x = sender(z)
| . . |         & (z
| . . |            λe,e'. e <loc e'
| . . |            ∨ isrcv(kind(e')) & e = sender(e')^n-1 y) fi e')
| . . |   ⇒
| . . |   time(e)<time(e'))  by Analyze 0 THEN SplitOnConclITE
| . . |
| . . |\
| . . | .....truecase .....
| . . | 1. the_w : World
| . . | 2. FairFifo
| . . | 3. ∀e,z:E.
| . . |    e <loc z ∨ isrcv(kind(z)) & e = sender(z)
| . . |    ⇒
| . . |    time(e)<time(z)
| . . | 4. n : ℤ
| . . | 5. 0<n
| . . | 6. 0<n-1
| . . |    ⇒
| . . |    (∀e,e':E.
| . . |     (e
| . . |      λe,e'. e <loc e' ∨ isrcv(kind(e')) & e = sender(e')^n
| . . |      -1 e')
| . . |     ⇒
| . . |     time(e)<time(e'))
| . . | 7. ¬n = 1
| . . | 8. 0<n
| . . | 9. n = 0
| . . | ⊢  ∀e,e':E. (e (λx,y. x = y) e') ⇒ time(e)<time(e')  by Auto
| . . |
| . . .\
| . . . .....falsecase .....
| . . . 1. the_w : World
| . . . 2. FairFifo
| . . . 3. ∀e,z:E.
| . . .    e <loc z ∨ isrcv(kind(z)) & e = sender(z)
| . . .    ⇒
| . . .    time(e)<time(z)
| . . . 4. n : ℤ
| . . . 5. 0<n
| . . . 6. 0<n-1
| . . .    ⇒
| . . .    (∀e,e':E.
| . . .     (e
| . . .      λe,e'. e <loc e' ∨ isrcv(kind(e')) & e = sender(e')^n
| . . .      -1 e')
| . . .     ⇒
| . . .     time(e)<time(e'))
| . . . 7. ¬n = 1
| . . . 8. 0<n
| . . . 9. ¬n=_2 0
| . . . ⊢  ∀e,e':E.
```

```
| . . .    (e
| . . .     (λx,y. ∃z:E.
| . . .      x <loc z ∨ isrcv(kind(z)) & x = sender(z)
| . . .      & (z
| . . .         λe,e'. e <loc e' ∨ isrcv(kind(e')) & e = sender(e')^n
| . . .         -1 y)) e')
| . . .    ⇒
| . . .    time(e)<time(e')
| . . .   by ((Analyze 6 THEN Analyze 0) THEN (Analyze 0)
| . . .       THEN
| . . .       (Reduce 0)
| . . .       THEN
| . . .       (Analyze 0))
| . . .      THEN
| . . .      ExRepD
| . . .      THEN
| . . .      AssertBY (time(e)<time(z)) (BackThru 3)
| . . .      THEN
| . . .      Assert (time(z)<time(e'))
| . . .      THEN
| . . .      BackThruSomeHyp
|
.\
. 1. the_w : World
. 2. e : E
. 3. e' : E
. 4. FairFifo

. 5. ∃n:ℕ+. e λe,e'. e <loc e' ∨ isrcv(kind(e')) & e = sender(e')^n e'
. 6. ∀n:ℕ+, e,e':E.
.    (e λe,e'. e <loc e' ∨ isrcv(kind(e')) & e = sender(e')^n e')
.    ⇒
.    time(e)<time(e')
. ⊢  time(e)<time(e')  by ExRepD THEN InstHyp [n;e;e'] -1
END
```