combine-ecl-tuples2[11,40]

```
combine-ecl-tuples2(A; B; f; g)
≡def spreadn(A;
             Ta,ksa,ia,ga,ha,aa,ea.spreadn
             (B;
              Tb,ksb,ib,gb,hb,ab,eb.<:Ta × (:Tb × (?𝔹))
              , append(ksa; ksb)
              , <ia, ib, inr · >
              , λk′,s,v,x. let sa = if deq-member(Kind-deq; k′; ksa)
                               then ga(k′,s,v,x.1)
                               else x.1
                               fi in
                               let sb = if deq-member(Kind-deq; k′; ksb)
                                   then gb(k′,s,v,(x.2).1)
                                   else (x.2).1
                                   fi in
                                   <sa
                                   , sb
                                   , combine-halt-info(ea;
                                                       eb;
                                                       (λm.ha(m,sa));
                                                       (λm.hb(m,sb));
                                                       (x.2.2))>
              , λn,x. f((x.2.2),λm.ha(m,x.1),λm.hb(m,(x.2).1),n)
              , λn,k′,s,v,x. g
                             (ha(0,x.1)
                             ,hb(0,(x.2).1)
                             ,reduce((λm,b. bor((ha(m,x.1)); b)); ff; ea)
                             ,reduce((λm,b. bor((hb(m,(x.2).1)); b)); ff; eb)
                             ,if deq-member(Kind-deq; k′; ksa)
                              then aa(n,k′,s,v,x.1)
                              else ff
                              fi
                             ,if deq-member(Kind-deq; k′; ksb)
                              then ab(n,k′,s,v,(x.2).1)
                              else ff
                              fi )
              , merge(ea; eb)>))
```

clarification:

```
combine-ecl-tuples2(A; B; f; g)
≡def spreadn(A;
```

```
Ta,ksa,ia,ga,ha,aa,ea.spreadn
(B;
 Tb,ksb,ib,gb,hb,ab,eb.<:Ta × (:Tb × (𝔹 + Unit))
 , append(ksa; ksb)
 , <ia, ib, inr · >
 , λk′,s,v,x. let sa = if deq-member(Kind-deq; k′; ksa)
                  then ga(k′,s,v,x.1)
                  else x.1
                  fi in
                   let sb = if deq-member(Kind-deq; k′; ksb)
                       then gb(k′,s,v,(x.2).1)
                       else (x.2).1
                       fi in
                        <sa
                       , sb
                       , combine-halt-info(ea;
                                           eb;
                                           (λm.ha(m,sa));
                                           (λm.hb(m,sb));
                                           (x.2.2))>
 , λn,x. f((x.2.2),λm.ha(m,x.1),λm.hb(m,(x.2).1),n)
 , λn,k′,s,v,x. g
                (ha(0,x.1)
                ,hb(0,(x.2).1)
                ,reduce((λm,b. bor((ha(m,x.1)); b)); ff; ea)
                ,reduce((λm,b. bor((hb(m,(x.2).1)); b)); ff; eb)
                ,if deq-member(Kind-deq; k′; ksa)
                 then aa(n,k′,s,v,x.1)
                 else ff
                 fi
                ,if deq-member(Kind-deq; k′; ksb)
                 then ab(n,k′,s,v,(x.2).1)
                 else ff
                 fi )
 , merge(ea; eb)>))
```

http://www.nuprl.org/FDLcontent/p0_963683_/p51_74682_{combine-ecl-tuples2}.html