The Happens-Before Relationship

Here is a short summary of the definitions involved in the Happens-Before relation.

1. A trace is a sequence of events.

   \[
   \text{Events } E ::= \text{start}(T) \\
   | \text{end}(T) \\
   | \text{read}(T,x,v) \\
   | \text{write}(T,x,v) \\
   | \text{lock}(T,x) \\
   | \text{unlock}(T,x)
   \]

2. Let \( E_1 < E_2 \) be the ordering of events as they appear in the trace.

3. Define happens-before ordering \( <: \) in a trace \( R \) as follows:
   \( E_1 <: E_2 \) iff \( E_1 < E_2 \) and one of the following holds:
   a) \( \text{thread}(E_1) = \text{thread}(E_2) \)
   b) \( E_1 \) is unlock(T1,x) and \( E_2 \) lock(T2,x)
   c) there exists \( E_3 \) with \( E_1 <: E_3 \) and \( E_3 <: E_2 \)

4. Updates are visible based on the following rules. For a trace \( r \)
   containing \( EW == \text{write}(T_1,x,v_1) \) and \( ER == \text{read}(T_2,x,v_2) \):

   \( EW \) "is not visible" to \( ER \) if
   - \( ER <: EW \)
   - There exists some event \( EW_2 == \text{write}(T,x,v_3) \) such that \( EW <: EW_2 <: R \)

   Otherwise \( EW \) is visible at \( ER \)

5. A data race takes place when there are two events in trace \( R \) that
   - access the same memory location
   - at least one is a write
   - they are unordered according to happens-before