



WIKIPEDIA
The Free Encyclopedia

[Main page](#)
[Contents](#)
[Featured content](#)
[Current events](#)
[Random article](#)
[Donate to Wikipedia](#)
[Wikipedia store](#)

Interaction
[Help](#)
[About Wikipedia](#)
[Community portal](#)
[Recent changes](#)
[Contact page](#)

Tools
[What links here](#)
[Related changes](#)
[Upload file](#)
[Special pages](#)
[Permanent link](#)
[Page information](#)
[Wikidata item](#)
[Cite this page](#)


Print/export
[Create a book](#)
[Download as PDF](#)
[Printable version](#)

Languages 
[Deutsch](#)
[Français](#)
[Nederlands](#)
 [Edit links](#)

[Create account](#) [Log in](#)


Article [Talk](#)

[Read](#) [Edit](#) [More](#) ▾



Happened-before

From Wikipedia, the free encyclopedia
(Redirected from [Lamport ordering](#))

In [computer science](#), the **happened-before relation** (denoted: \rightarrow) is a relation between the result of two events, such that if one event should happen before another event, the result must reflect that, even if those events are in reality executed out of order (usually to optimize program flow). This involves [ordering](#) events based on the potential [causal relationship](#) of pairs of events in a concurrent system, especially [asynchronous distributed systems](#). It was formulated by [Leslie Lamport](#).^[1] In Java specifically, a **happens-before** relationship is a guarantee that memory written to by statement A is visible to statement B, that is, that statement A completes its write before statement B starts its read.^[1] 

The happened-before relation is formally defined as the least [strict partial order](#) on events such that:

- If events *a* and *b* occur on the same process, *a* \rightarrow *b* if the occurrence of event *a* preceded the occurrence of event *b*.
- If event *a* is the sending of a message and event *b* is the reception of the message sent in event *a*, *a* \rightarrow *b*

If there are other causal relationships between events in a given system, such as between the creation of a process and its first event, these relationships are also added to the definition.

Like all strict partial orders, the happened-before relation is [transitive](#), [irreflexive](#) and [antisymmetric](#), i.e.:

- $\forall a, b, c$, if *a* \rightarrow *b* and *b* \rightarrow *c*, then *a* \rightarrow *c* (transitivity) ;
- $\forall a, a \nrightarrow a$ (irreflexivity) ;
- $\forall a, b$, if *a* \rightarrow *b* \wedge *b* \rightarrow *a* then *a* = *b* (antisymmetry).


Because the happened-before relation is both irreflexive and antisymmetric, it follows that: if *a* \rightarrow *b* then *b* \nrightarrow *a*

The processes that make up a distributed system have no knowledge of the happened-before relation unless they use a [logical clock](#), like a [Lamport clock](#) or a [vector clock](#). This allows one to design algorithms for [mutual exclusion](#), and tasks like debugging or optimising distributed systems.

See also

- [Java Memory Model](#)
- [Lamport timestamps](#)

References

- ↑ Lamport, Leslie (1978). "Time, Clocks and the Ordering of Events in a Distributed System" , *Communications of the ACM*, 21(7), 558-565.

Categories: [Distributed computing problems](#)

This page was last modified on 2 September 2015, at 17:38.

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. By using this site, you agree to the [Terms of Use](#) and [Privacy Policy](#). Wikipedia® is a registered trademark of the [Wikimedia Foundation, Inc.](#), a non-profit organization.

[Privacy policy](#) [About Wikipedia](#) [Disclaimers](#) [Contact Wikipedia](#) [Developers](#) [Mobile view](#)

