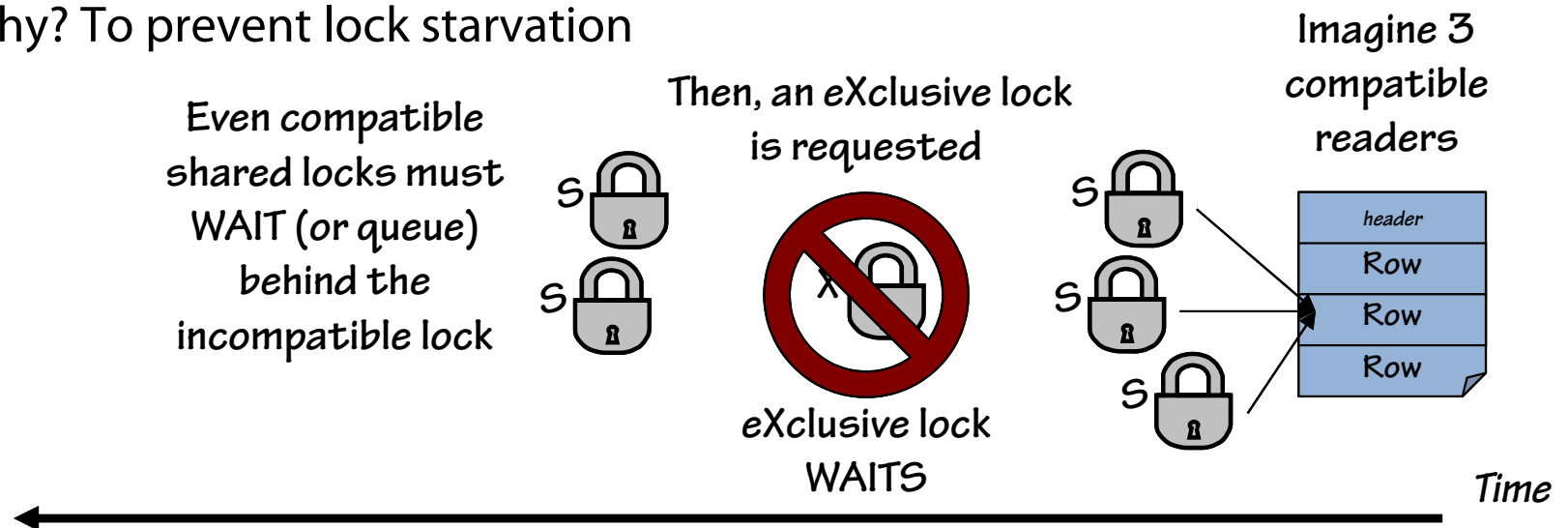


Blocking – Is it Really a Problem? It depends...

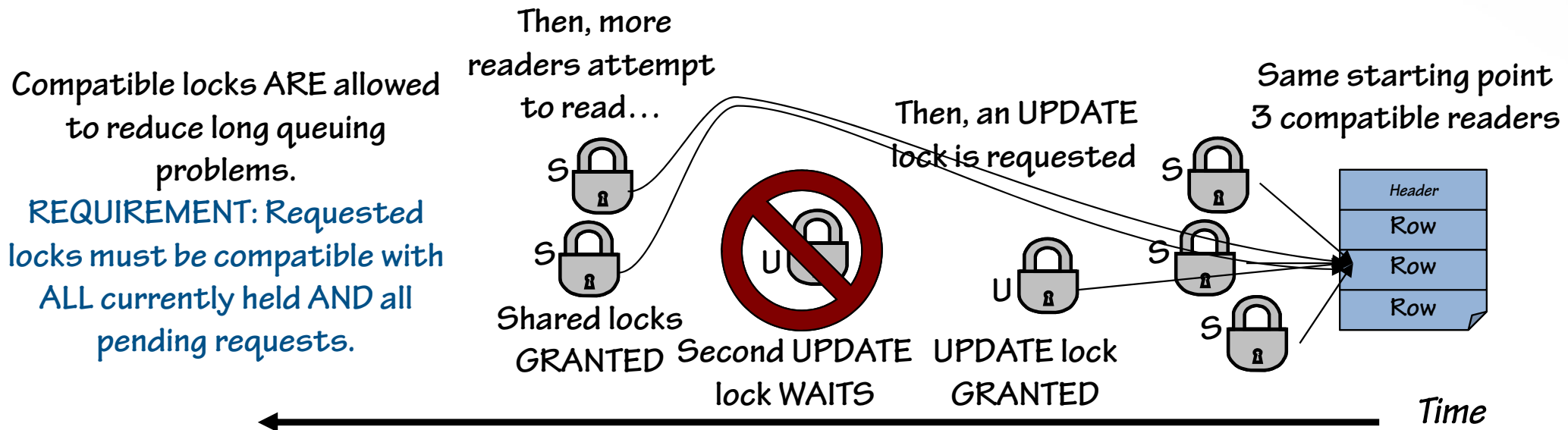
- Locks guarantee consistency
- First incompatible lock request waits
- Once an incompatible lock request is made then pending locks will wait even if they're compatible with locks currently held
 - Special case WHEN the requested lock IS compatible WITH ALL pending requests...
- Primary reason for the locks to WAIT
 - Why? To prevent lock starvation



Relaxed FIFO

To Reduce Really Long Blocking Chains

- <http://blogs.msdn.com/b/psssql/archive/2009/06/02/sql-server-lock-manager-and-relaxed-fifo.aspx> (<http://bit.ly/NWAgVC>)
- An update is incompatible with another update
- The requested (second) update will wait behind the current update...
- What about shared locks if ONLY update locks?
 - ALLOWED



Nasty Blocking CHAINS

Relaxed FIFO isn't Always Enough

- Imagine someone's running a NOLOCK query against a very large table (the query takes 4 minutes to run [ok, not that nasty])
- This query's running off-hours but the standard is to use NOLOCK
- An `sp_recompile tname` is requested

Unfortunately now...
everybody waits.....

Relaxed FIFO only let's through the
folks that are compatible with ALL
pending locks (not just those that
are currently held).

This can create terribly long blocking
chains.

