**Synchrone Kommunikation**

- **Synchrone Kommunikation:**
  “send” und “receive” geschehen **virtuell gleichzeitig**

- **Sender** ist **blockiert**, bis er vom Empfang seiner Nachricht erfährt
  - Sendezeitpunkt ist innerhalb des Blockadeintervalls beliebig **verschiebbar**
  - als wäre der Sender vor und nach dem virtuellen Sendezeitpunkt “idle”

---

**Modeling Synchronous Communication**

- **How do we define** distributed computations with synchronous message passing?
  - or: **characterize** those distributed computations that can be **realized** with synchronous communications?

- **Proposition:**
  Synchronous = virtually simultaneous
  = as if msg transmission were instantaneous
  
  - But: aren’t instantaneous message transmissions unrealistic?

- **Can we always** apply a suitable rubber band transformation such that all message arrows become vertical?

---

**Konsequenz:**
Man darf so tun, als wären Nachrichten nie unterwegs!

- “blitzschnelle” Nachrichten
- vereinfacht viele Argumentationen
- formale Unterscheidung synchron / asynchron durch Kausalrelation (synchron: Abhängigkeit des Senders vom Empfänger)
“As if” Messages were Instantaneous?

If for a distributed computation a phenomenon can be observed which is impossible with instantaneous messages, the computation must not be realizable with synchronous message passing semantics

$$\Rightarrow$$ message passing should then not be called “synchronous”

**Example:**

The observer first asks A about the number of messages it sent to B

Then it asks B about the number of messages it received from A

- The message from A to B is *overtaken in an indirect way by a chain of other messages*
- The direct message can therefore *not be made vertical* by a rubber band transformation
  (A message of the chain would then go backwards in time)

- Another computation which is not possible with synchronous communications ($\Rightarrow$ deadlock):

  Although each single arrow can be made vertical, it is not possible to draw the diagram in such a way that both arrows are vertical!

Observer learns that a message from A to B is *in transit* for a certain *duration* $\Rightarrow$ not synchronous!
Various Characterizations of Synchronous Communications

- Question: are they all equivalent?
- Note: some characterizations are informal or less formal than others

1) Best possible approximation of instantaneous communications
(i.e., without clocks, it is not possible to prove that a message was not transmitted instantaneously)

2) Space-time diagrams can be drawn such that all message arrows are vertical

3) Communication channels always appear to be empty
(i.e., messages are never seen to be in transit)

4) Corresponding send-receive events form one single atomic action

- But what exactly does “atomic” mean?
- Does the combined event happen before or after the wave? Should this be possible with synchronous communication?

5) \( \exists \) linear extension of \((E, <)\) such that \( \forall \) corresponding communic. events \( s, r: r \) is an immediate predecessor of \( s \)

- The example has 4 different linearizations: in all of them a pair of corresponding send-receive events is separated by other events - hence this computation cannot be realized synchronously
- Motivation: corresponding events form a single atomic action

6) Define a (transitive) scheduling relation ‘<‘ on messages:
\[ m \lessdot n \iff \text{send}(m) < \text{receive}(n) \]
The graph of ‘<‘ must be cycle-free

- Then whole messages (i.e., corresponding send-receive events \( s, r \)) can be scheduled at once (\( s \) before \( r \)), otherwise this is not possible
7) No cycle is possible by moving along message arrows in either direction, but always from left to right on process lines.

- Interpretation: ignoring the direction of message arrows ==> send / receive is "symmetric"
- "identify" send / receive
- If such a cycle exists ==> no "first" message to schedule
- If no such cycle does exists ==> message schedule exists

8) Synchronous causality relation $\ll$ is a partial order

Definition of $\ll$:

1. If a before b on the same process, then a $\ll$ b
2. $x \ll s$ iff $x \ll r$ ("common past")
3. $s \ll x$ iff $r \ll x$ ("common future")
4. Transitive closure

Interpretation: corresponding s, r are not related, but with respect to the synchronous causality relation they are "identified"

Example:

- Compare this characterization to characterization 6
- Why is the definition of $\ll$ sensible?

Further reading (for those who are interested): Charron-Bost, Mattern, Tel: Synchronous, Asynchronous and Causally Ordered Communication. Distributed Computing, Vol. 9 No. 4, pp. 173 - 191, 1996
http://www.informatik.tu-darmstadt.de/VS/Publikationen/papers/syn_asy.ps