



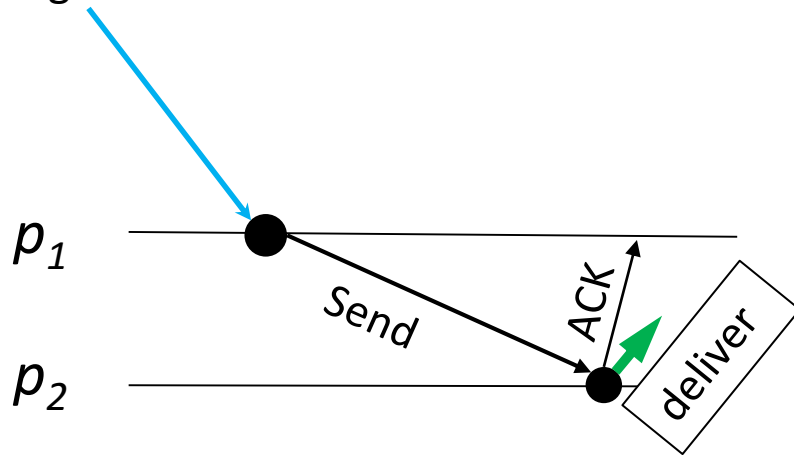
Basic Abstraction

vanilladb.org

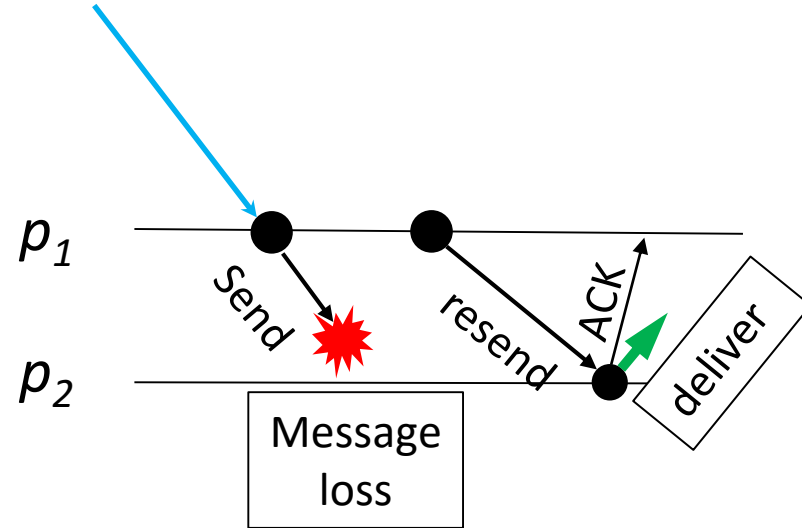
Perfect Point to Point Link

- How to cope with message loss?
 - Message retransmission and eliminating duplicates

Message to be sent



Message to be sent



Perfect Point to Point Link

- Properties
 - **Reliable delivery:** if neither the sender nor the receiver crashes, then the receiver eventually delivers a message sent by the sender
 - Keep retransmitting the message until an ACK is received
 - **No duplication:** a receiver may receive a message many times, but can only deliver it once
 - Sequence number
 - **No creation:** if a message is delivered, it must be sent by some process
 - Checksum

Perfect Point to Point Link

Algorithm 2.1 Retransmit Forever

Implements:

StubbornPointToPointLink (sp2p).

Uses:

FairLossPointToPointLinks (flp2p).

upon event $\langle \text{Init} \rangle$ do

sent := \emptyset ;
startTimer (TimeDelay);

upon event $\langle \text{Timeout} \rangle$ do

forall $(\text{dest}, m) \in \text{sent}$ do
trigger $\langle \text{flp2pSend} \mid \text{dest}, m \rangle$;
startTimer (TimeDelay);

upon event $\langle \text{sp2pSend} \mid \text{dest}, m \rangle$ do

trigger $\langle \text{flp2pSend} \mid \text{dest}, m \rangle$;
sent := sent \cup $\{(\text{dest}, m)\}$;

upon event $\langle \text{flp2pDeliver} \mid \text{src}, m \rangle$ do

trigger $\langle \text{sp2pDeliver} \mid \text{src}, m \rangle$;

Algorithm 2.2 Eliminate Duplicates

Implements:

PerfectPointToPointLinks (pp2p).

Uses:

StubbornPointToPointLinks (sp2p).

upon event $\langle \text{Init} \rangle$ do

delivered := \emptyset ;

upon event $\langle \text{pp2pSend} \mid \text{dest}, m \rangle$ do

trigger $\langle \text{sp2pSend} \mid \text{dest}, m \rangle$;

upon event $\langle \text{sp2pDeliver} \mid \text{src}, m \rangle$ do

if $(m \notin \text{delivered})$ then
delivered := delivered \cup $\{ m \}$;
trigger $\langle \text{pp2pDeliver} \mid \text{src}, m \rangle$;



Perfect Failure Detection

- How to detect a node failure?
 - Detect timeout for *heartbeats*
 - If not receiving a heartbeat from a process p for a long time, then deem p has crashed

Perfect Failure Detection

- Uses:
 - *PerfectPointToPointLink*
- Properties
 - **Strong completeness:** eventually every correct process knows which processes are still alive.
 - Achieved by broadcasting which nodes are failed, or everyone can detect by themselves
 - **Strong accuracy:** if a process p is detected by any process, then p has crashed
 - A process is detected as failure iff it has crashed

Perfect Failure Detection

Algorithm 2.4 Exclude on Timeout

Implements:

PerfectFailureDetector (\mathcal{P}).

Uses:

PerfectPointToPointLinks (pp2p).

upon event $\langle \text{Init} \rangle$ **do**

 alive := Π ;
 detected := \emptyset ;
 startTimer (TimeDelay);

upon event $\langle \text{Timeout} \rangle$ **do**

 forall $p_i \in \Pi$ **do**
 if $(p_i \notin \text{alive}) \wedge (p_i \notin \text{detected})$ **then**
 detected := detected $\cup \{ p_i \}$;
 trigger $\langle \text{crash} \mid p_i \rangle$;
 trigger $\langle \text{pp2pSend} \mid p_i, [\text{HEARTBEAT}] \rangle$;
 alive := \emptyset ;
 startTimer (TimeDelay);

upon event $\langle \text{pp2pDeliver} \mid \text{src}, [\text{HEARTBEAT}] \rangle$ **do**

 alive := alive $\cup \{ \text{src} \}$;

