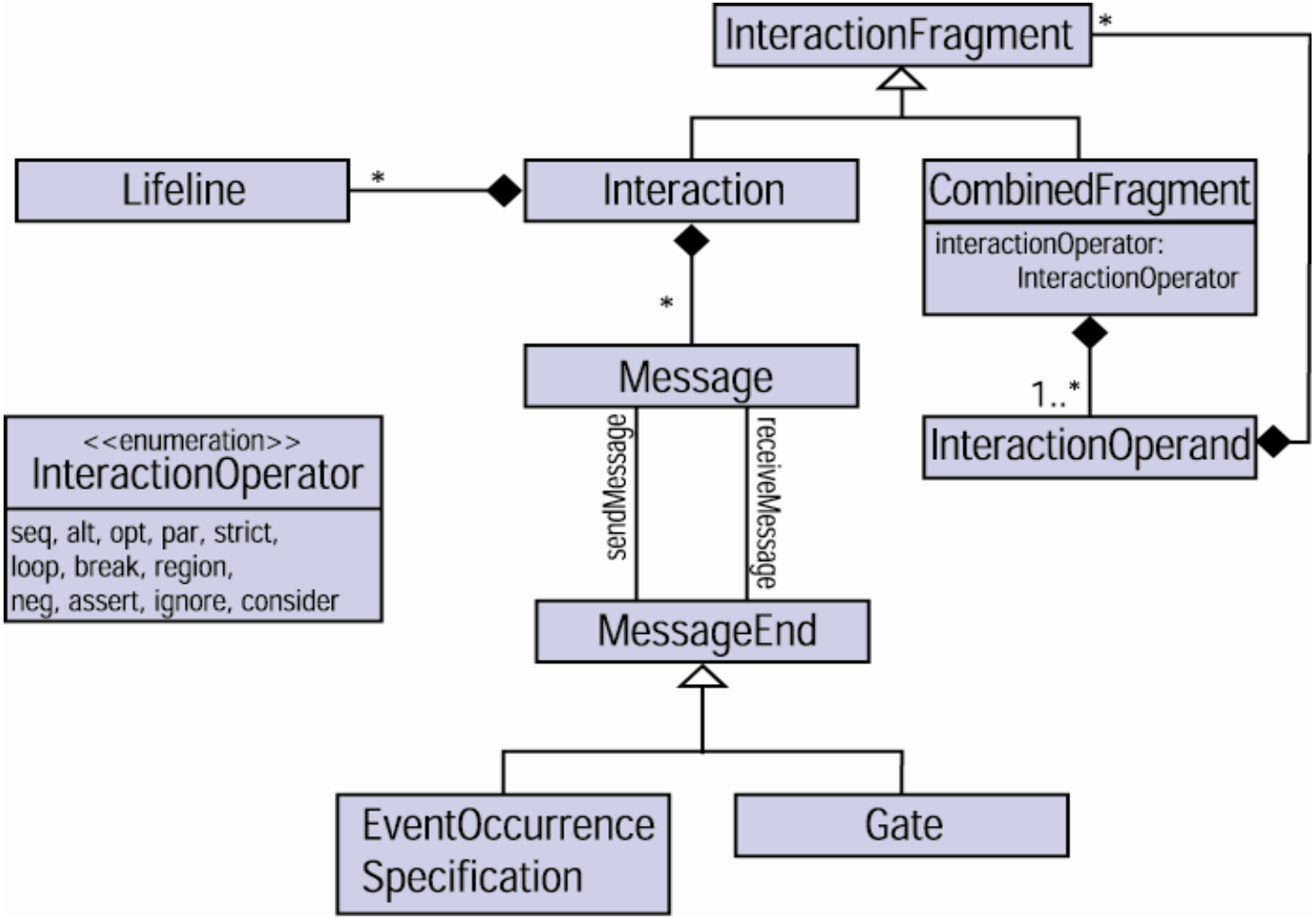
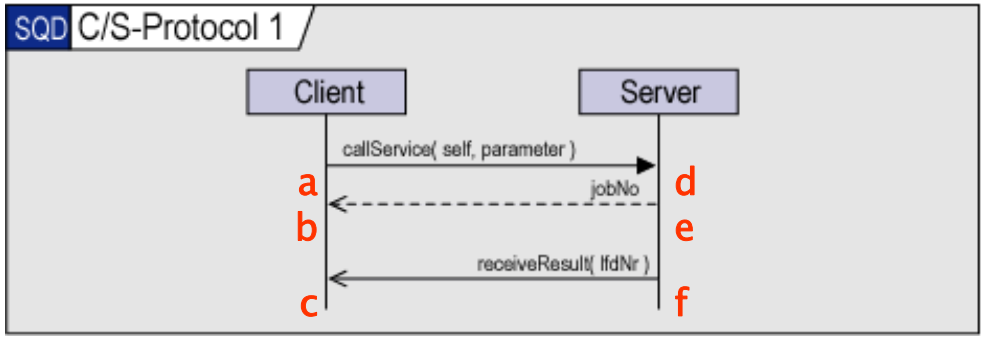
Main concepts (metamodel)



Semantics

- The meaning of an interaction is
  - a set of valid traces, plus
  - a set of invalid traces.
- Traces are made up of occurrences of events such as
  - sending/receiving a message,
  - instantiating/terminating an object, or
  - time/state change events.
- Two types of constraints determine the valid traces:
  1) send occurs before receive,
  2) order on lifelines is definite.



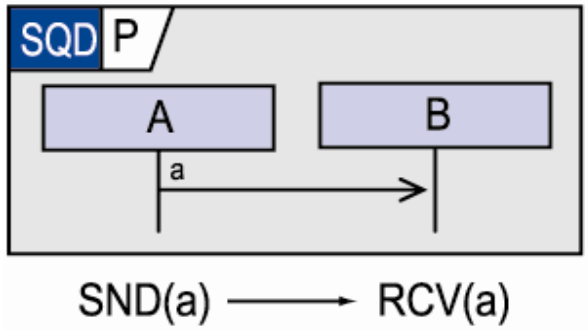
This diagram contains the following seven constraints:

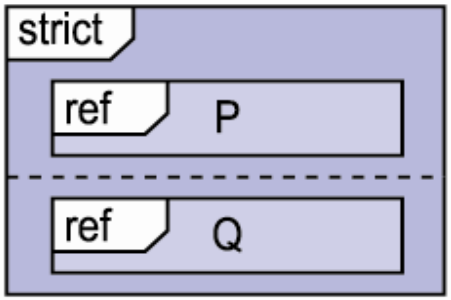
1) a→d, e→b, f→c
2) a→b, b→c, d→e, e→f

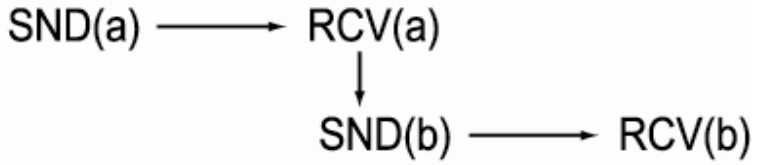
The set of resulting traces is:
{ a.d.e.b.f.c, a.d.e.f.b.c }.

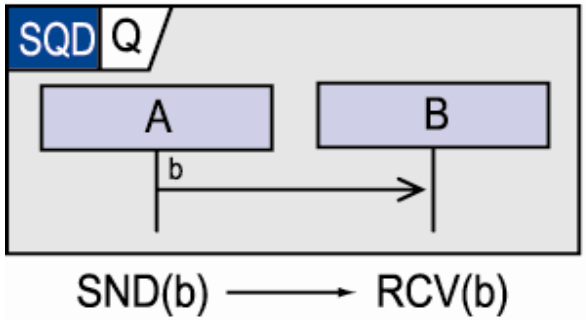
Interaction operators seq & strict

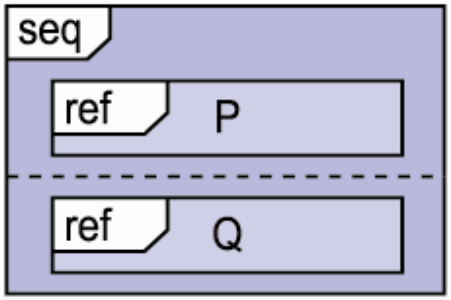
- **seq**
  - compose two interactions sequentially lifeline-wise (default!)
- **strict**
  - compose two interactions sequentially diagram-wise

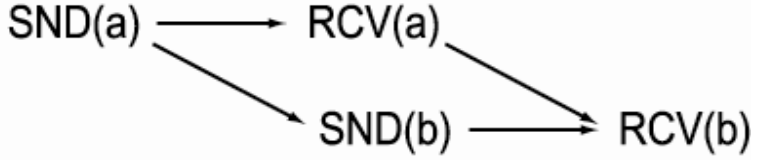












Interaction operator loop

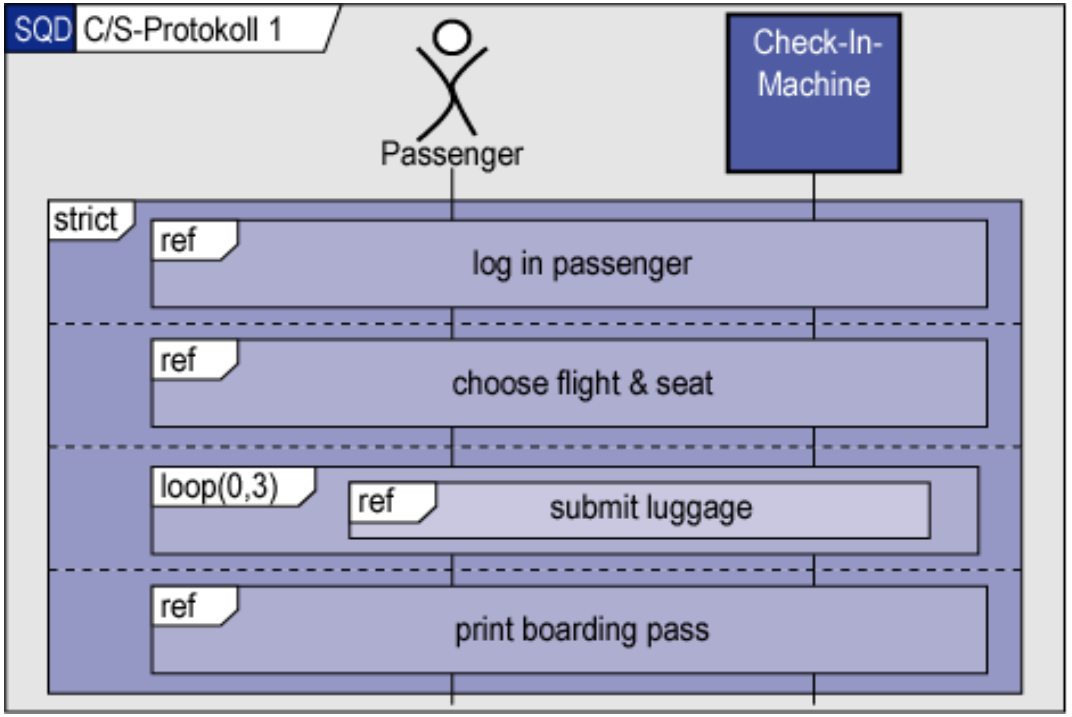
- **loop**
  - repeated application of seq

    loop(P, min, max) = seq(P, loop(P, min-1, max-1))

    loop(P, 0, max) = seq(opt(P), loop(P, 0, max-1))

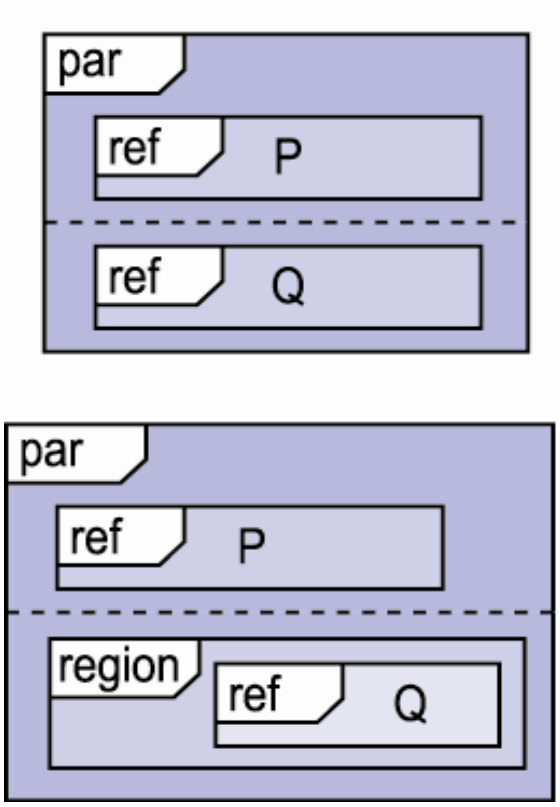
    loop(P, *) = seq(opt(P), loop(P, *))

    for some interaction fragment P



Interaction operators: interleaving

- **par**
  - shuffle arguments
- **region**
  - execute argument atomically, i.e. disallow interleaving



SND(a) ⟶ RCV(a)

SND(b) ⟶ RCV(b)

SND(a).RCV(a).SND(b).RCV(b)
SND(a).SND(b).RCV(a).RCV(b)
SND(a).SND(b).RCV(b).RCV(a)
SND(b).SND(a).RCV(a).RCV(b)
SND(b).SND(a).RCV(b).RCV(a)
SND(b).RCV(b).SND(a).RCV(a)

SND(a).RCV(a).SND(b).RCV(b)
SND(a).SND(b).RCV(b).RCV(a)
SND(b).RCV(b).SND(a).RCV(a)

Interaction operators alt, opt, brk: choice

- **alt**
  - alternative complete execution of one of two interaction fragments
- **opt**
  - optional complete execution of interaction fragment:
    opt(P) = alt(P, nop)
- **break**
  - execute interaction fragment partially, skip rest, and jump to surrounding fragment

Interaction operators: abstraction

- **ignore**, **consider**
  - dual way of expressing:
    - allow the ignorable messages (!) anywhere
    - present only those messages that are to be considered
    - ⟦ ignore(P,Z) ⟧ = shuffle(⟦P⟧ , Z*)