

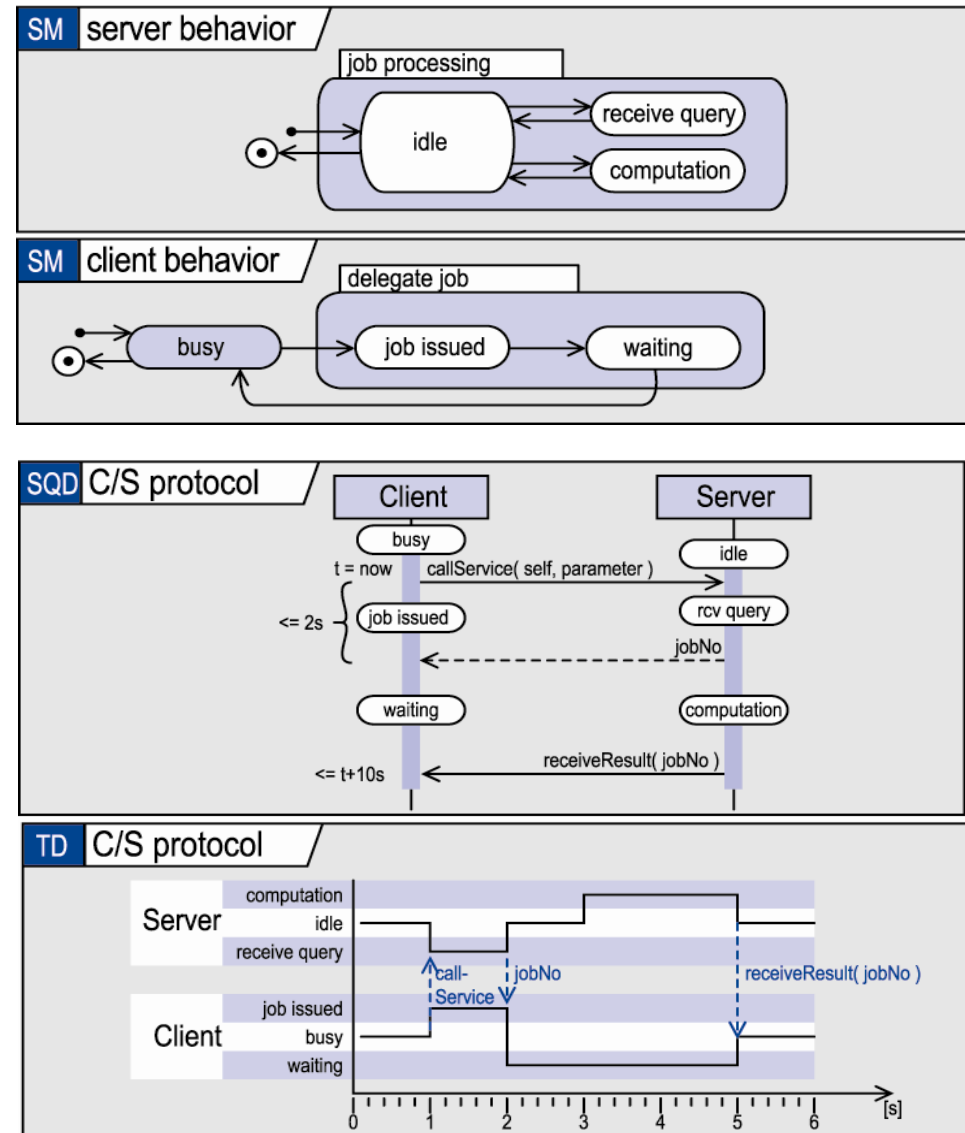
Usage: Test cases

- Like any other interaction, but with a different intention.
- Typically accompanied by a **tabular description** of purpose, expected parameters and result (similar to use case description).

identifier TF-AAA.CIA-4	name Check In (automatic) too much luggage
test goal If a passenger has too many pieces of luggage and tries to check in using the check in machine, he should be referred to the check in counter.	
precondition passenger is booked on respective flight	
arguments luggage, bonus mile card, booking data	
result passenger is referred to counter	
postcondition luggage is not checked in, passenger is checked in	
remarks, open questions none	

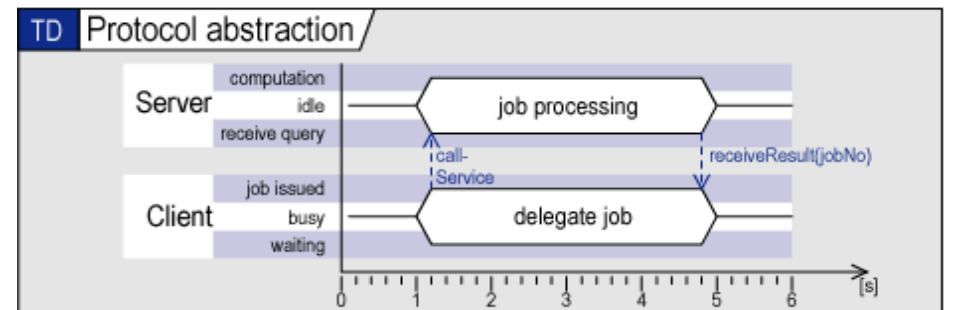
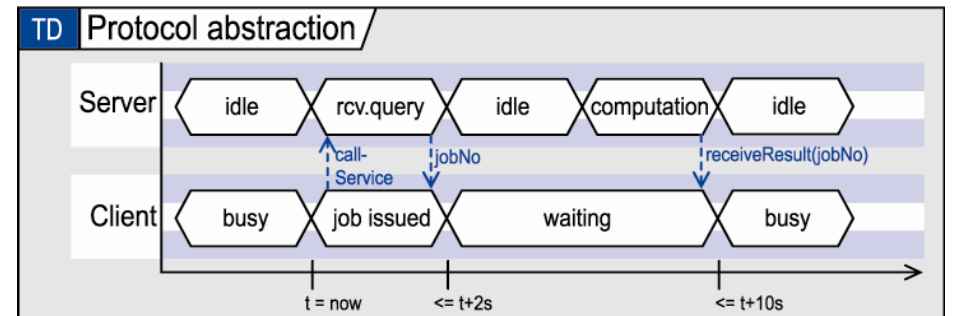
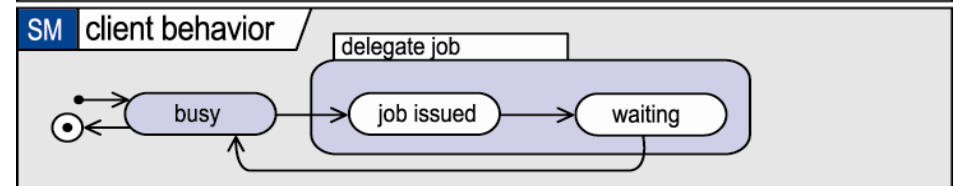
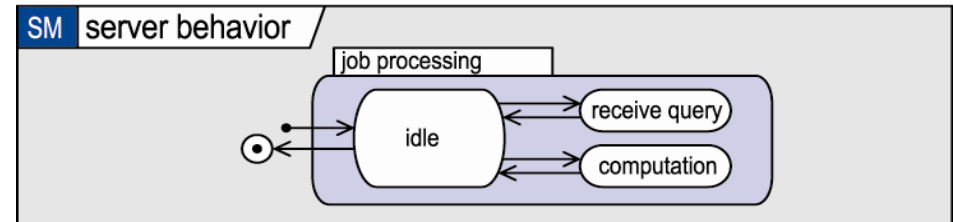
Usage: Timing specification

- For **embedded** and **real-time** systems, it may be important to specify absolute timings and state evolution over time.
- This is not readily expressed in sequence diagrams, much less communication diagrams.
- UML 2.0 introduces **timing diagrams** for this purpose.



Abstraction in timing diagram

- An alternative syntax presents states not on the vertical axis but as hexagons on the lifeline.
- Timing diagrams present the coordination of (the states of) several objects over (real) time.

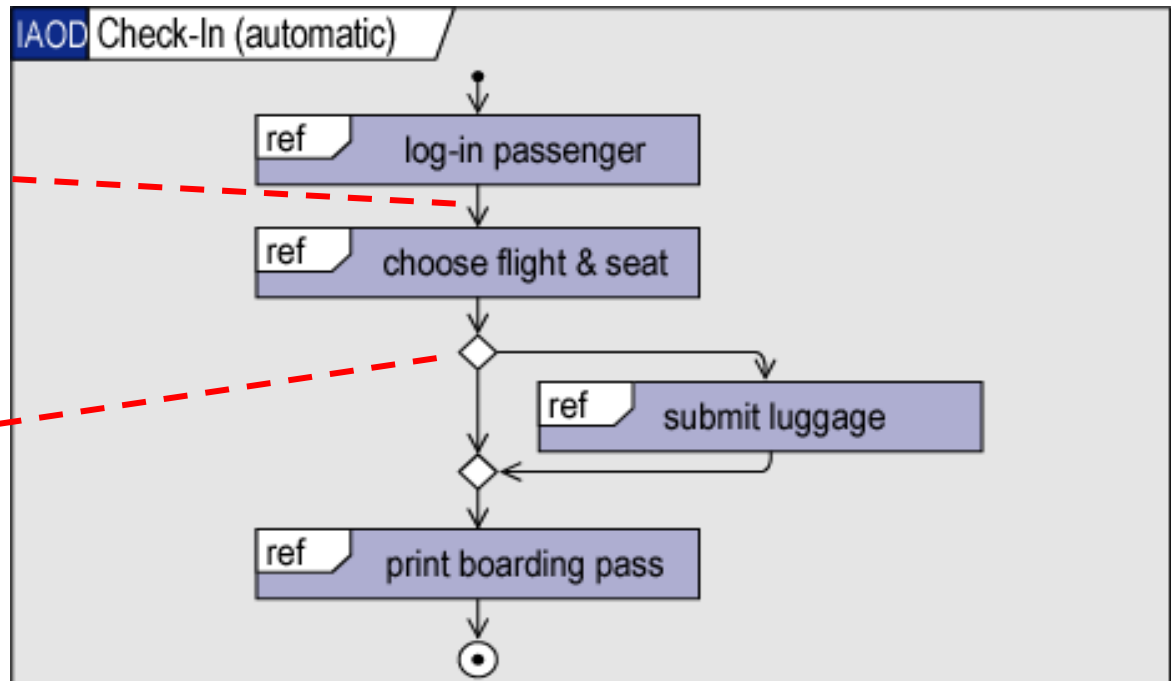


Usage: Interaction overview

- Organize large number of interactions in a more visual style
- Defined as equivalent to using interaction operators

sequence equivalent to seq

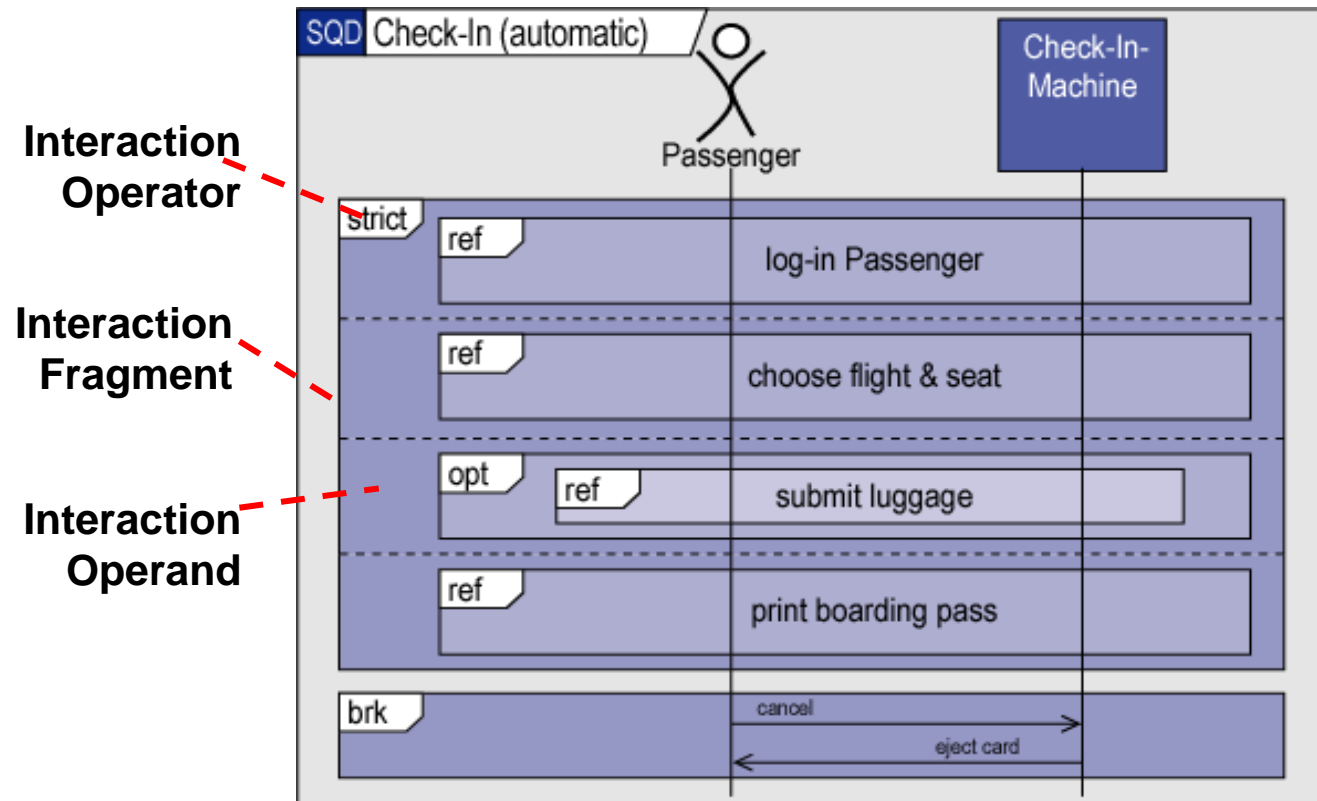
choice/merge
equivalent to alt/opt



also allowed: fork/join
(said to be equivalent to par, but ...)

Complex interactions

- A complex interaction is like a functional expression:
 - an InteractionOperator,
 - one or several InteractionOperands (separated by dashed lines),
 - (and sometimes also numbers or sets of signals).



Interaction operators (overview)

- strict
 - operand-wise sequencing
- seq
 - lifeline-wise sequencing
- loop
 - repeated seq
- par
 - interleaving of events
- region (aka. “critical”)
 - suspending interleaving
- consider
 - restrict model to specific messages
 - i.e. allow anything else anywhere
- ignore
 - dual to consider
- ref
 - macro-expansion of fragment
- alt
 - alternative execution
- opt
 - optional execution
 - syntactic sugar for alt
- break
 - abort execution
 - sometimes written as “brk”
- assert
 - remove uncertainty in specification
 - i.e. declare all traces as valid
- neg
 - declare all traces as invalid
(→ three-valued semantics)