Project – Background on Sequence Diagrams

Gunter Mussbacher
ECE, McGill University, Canada ◄► gunter.mussbacher@mcgill.ca

Based on material from: Bruegge & Dutoit, Lethbridge & Laganière, the Borland UML tutorial, K. Kostas, S. Somé, and D. Amyot

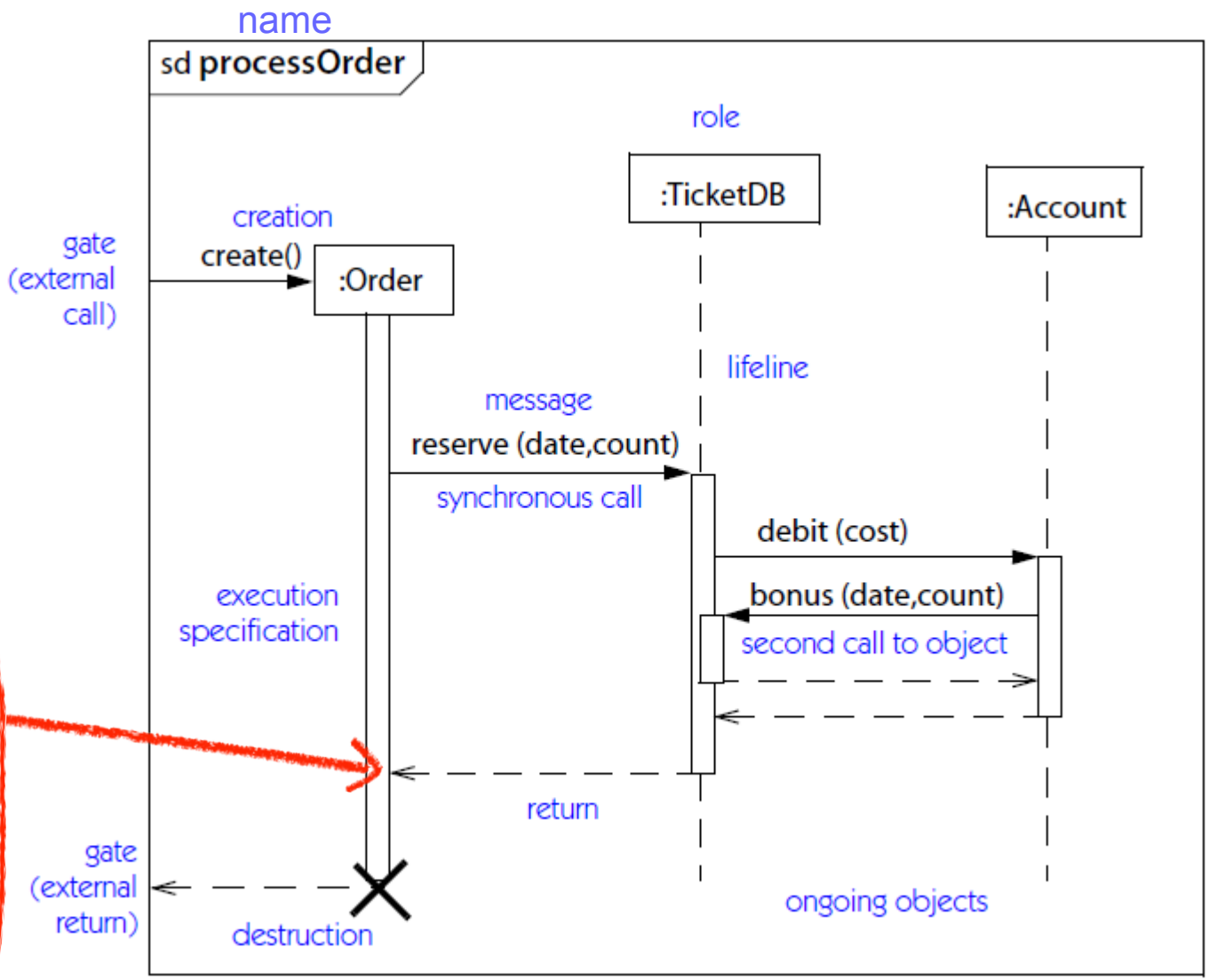
UML 2.x Sequence Diagrams

- Major improvements over first version of UML based on ITU-T's Message Sequence Charts

- The most important one: combined fragments
- Other improvements
  - (A)synchronous interactions
  - References
  - Hierarchical decomposition
  - Temporal aspects
  - …

Basic Notational Elements of Sequence Diagrams

- Describe the dynamic behavior as interactions between actors and the system and between objects of the system

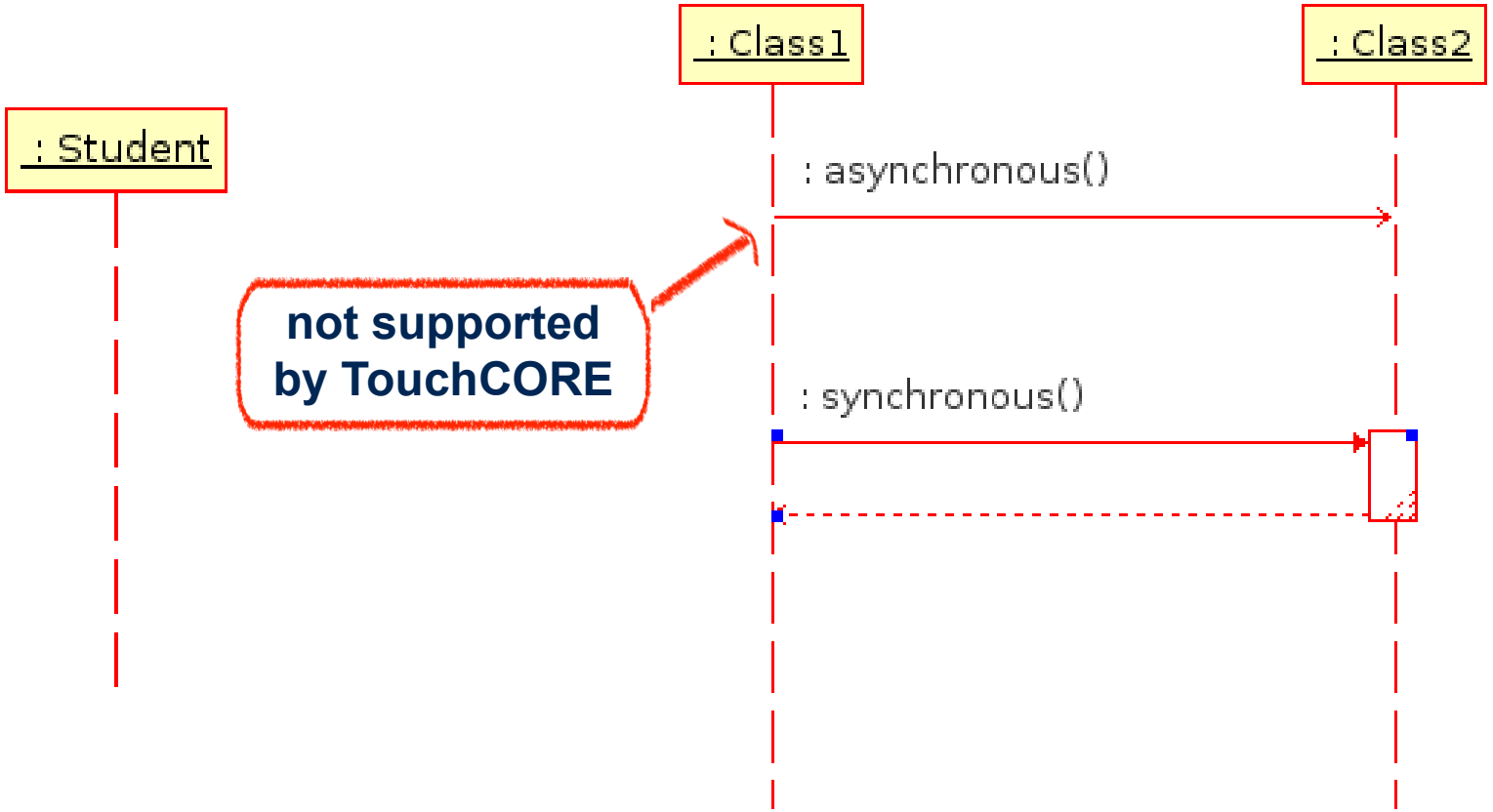


shown in TouchCORE only if method returns something and method behavior is not defined in its own message view

Source: UML Reference Manual

Lifelines and (A)synchronous Interactions

- Entities shown using lifeline participate in the interaction sequence by sending / receiving messages
- Messages can be synchronous or asynchronous



Combined Fragments

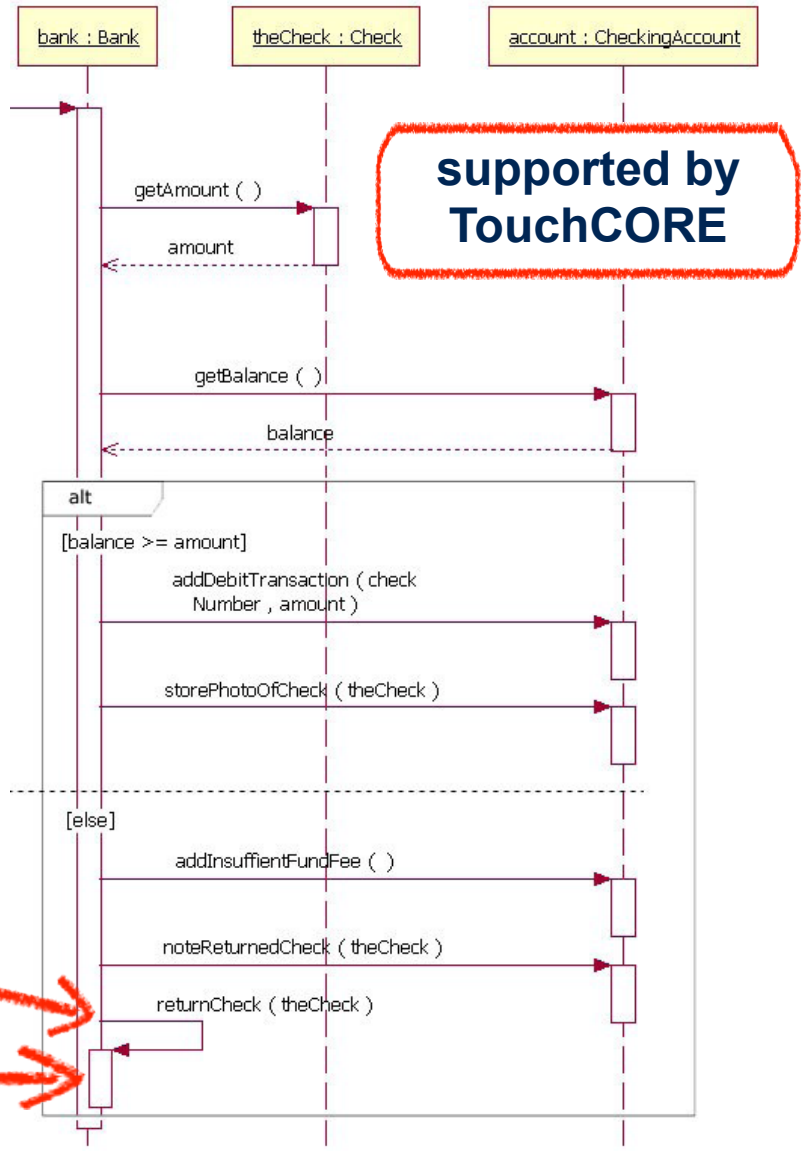
- Allow multiple sequences to be represented in compact form (may involve all scenario participants or just a subset)
- Combined fragment operators
  - alt, for alternatives with conditions
  - opt, for optional behavior
  - loop(lower bound, upper bound), for loops
  - par, for concurrent behavior
  - critical, for critical sections
  - break, to show a scenario will not be covered
  - assert, required
  - ignore/consider(list of messages), for filtering messages
  - neg, for invalid or mis-use scenarios that must not occur
  - strict or seq, for strict/weak sequencing
  - ref, for referencing other sequence diagrams

only these are supported by TouchCORE

TouchCORE also supports "disruptable" for try-catch blocks, which is not standard UML

Combined Fragments – Alternative

- Alternative (operator alt)
  - Multiple operands (separated by dashed lines)
  - Each operand has guard condition (no condition implies true)
  - One will be chosen exclusively – nondeterministic if more than one evaluates to true
  - Special guard: else
    - True if no other guard condition is true

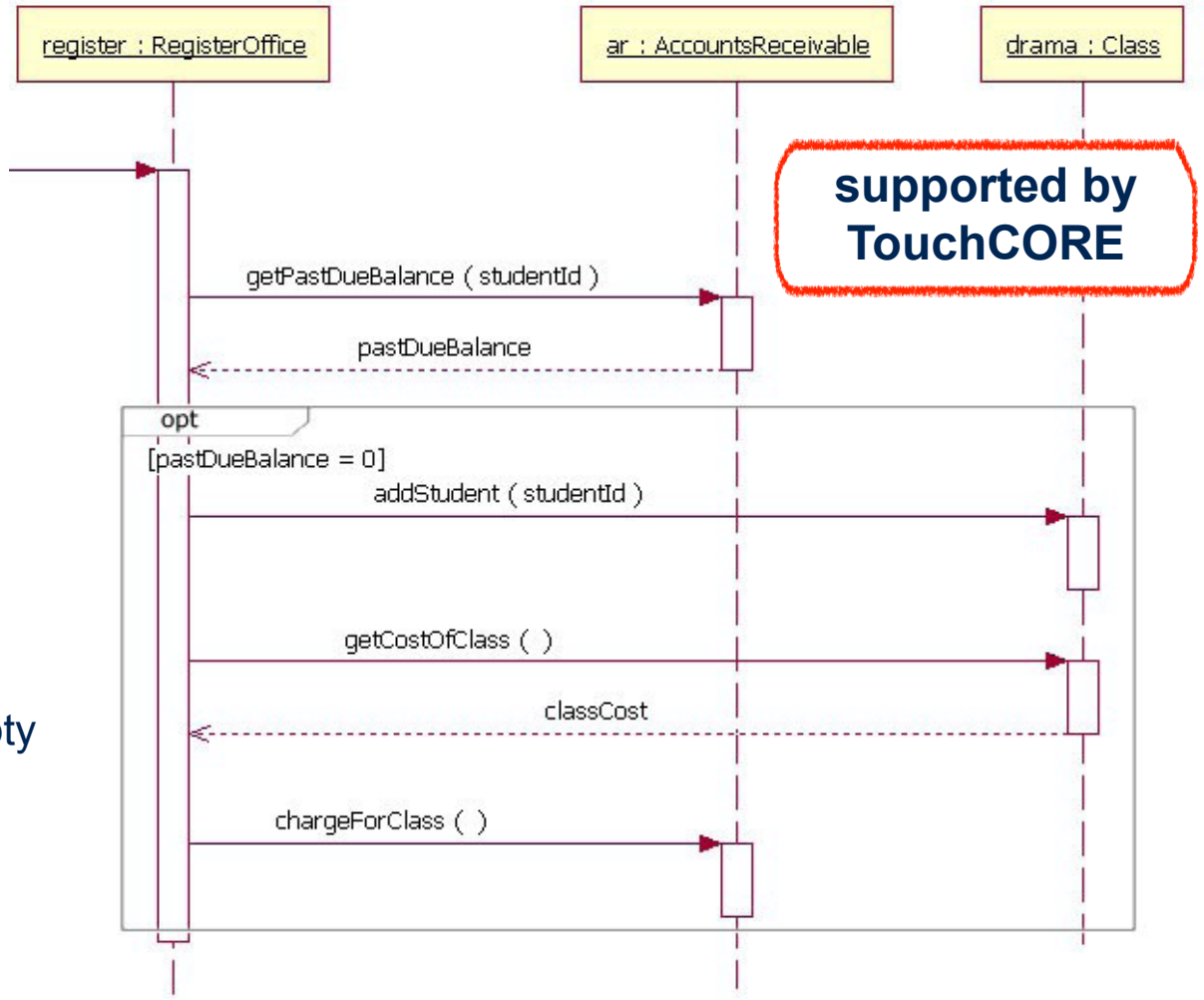


self message is also supported by TouchCORE

execution bar on top of another one not shown by TouchCORE

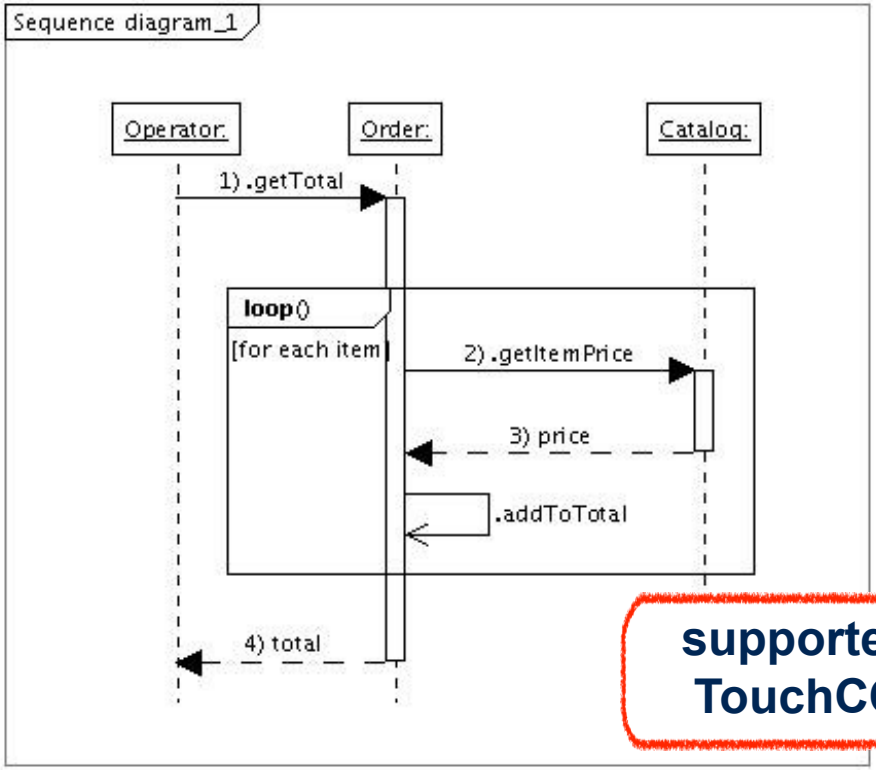
Combined Fragments – Optional

- Optional (operator opt)
  - To specify a guarded behavior fragment with no alternative
  - Special case of alt
  - Equivalent to an alt with two operands
    - The first is the same as the operand for the opt
    - The second is an empty operand with an else guard



Combined Fragments – Loop

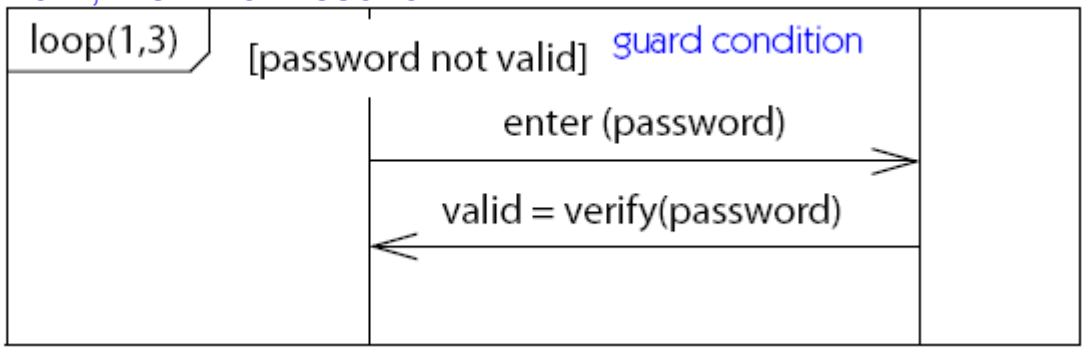
- Loop (operator loop)
  - Loop fragment may execute multiple times
  - At least executed the minimum count
  - Up to a maximum count as long as the guard condition is true (no condition implies true)



supported by TouchCORE

minimum, maximum count

Executes 1 to 3 times

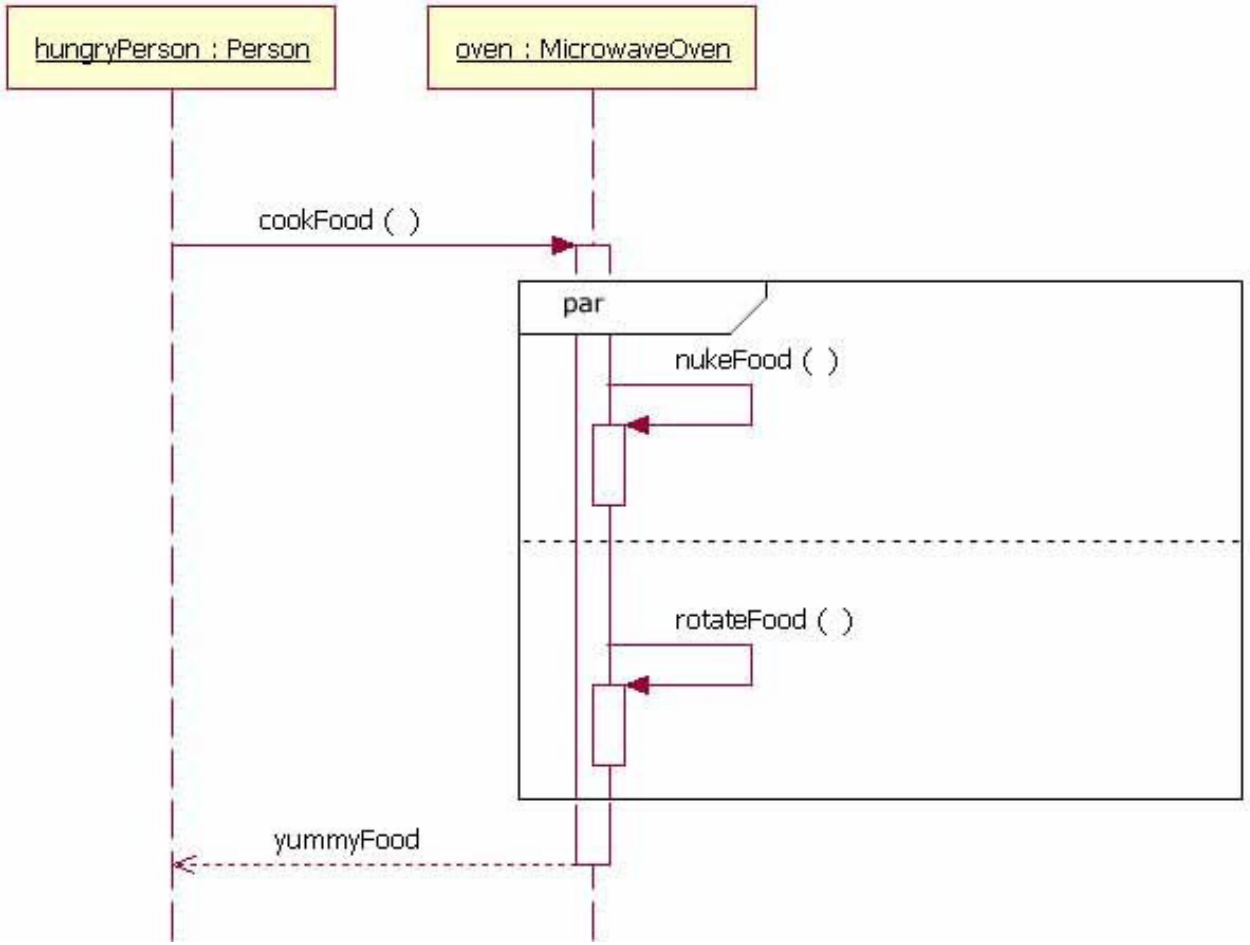


Source for Password Example: UML Reference Manual

Combined Fragments – Concurrency

- Concurrency (operator par)
  - Two or more operands that execute in parallel

not supported by TouchCORE



Nested Combined Fragments

supported by TouchCORE