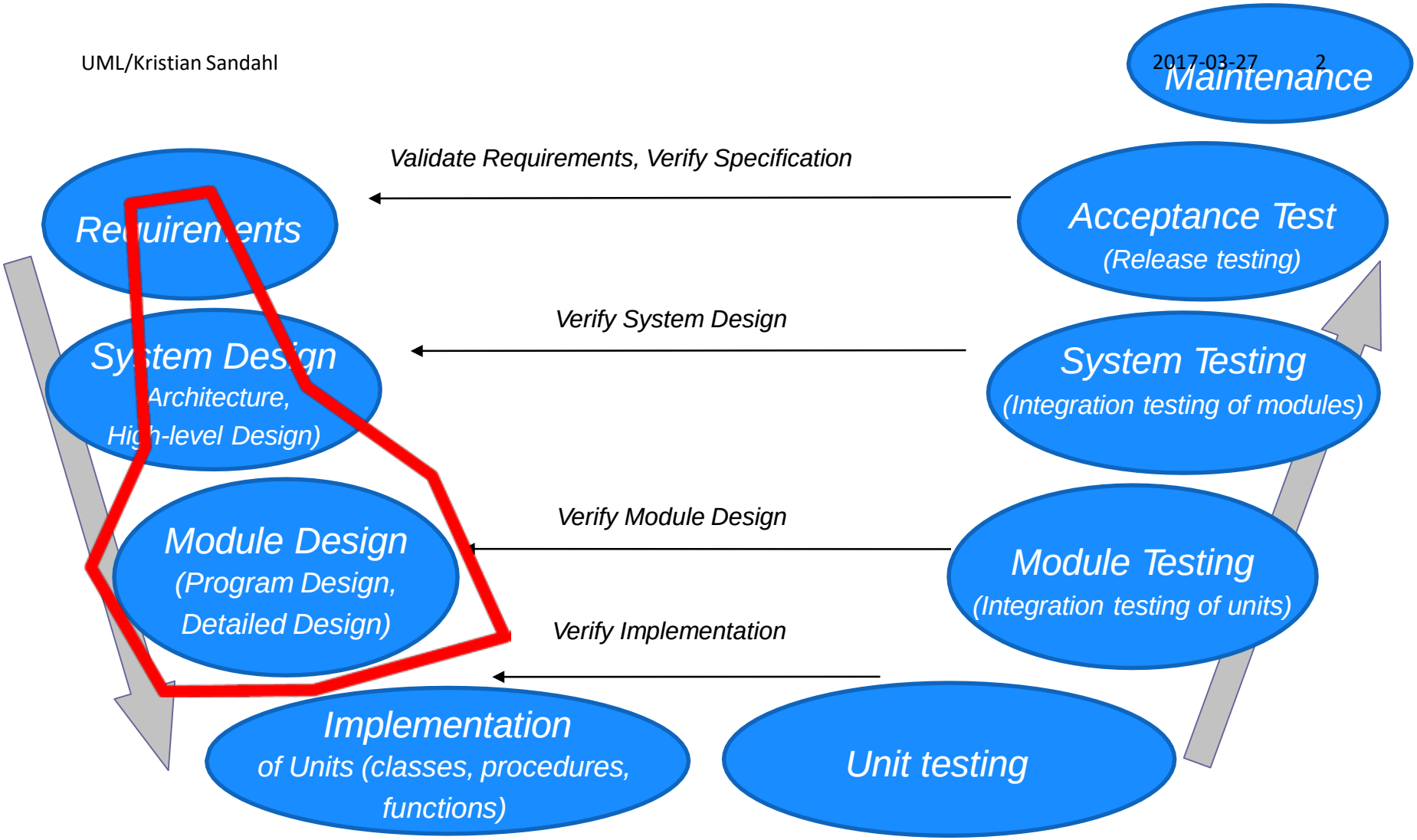
UML behavior models

Kristian Sandahl

LINKÖPING UNIVERSITY

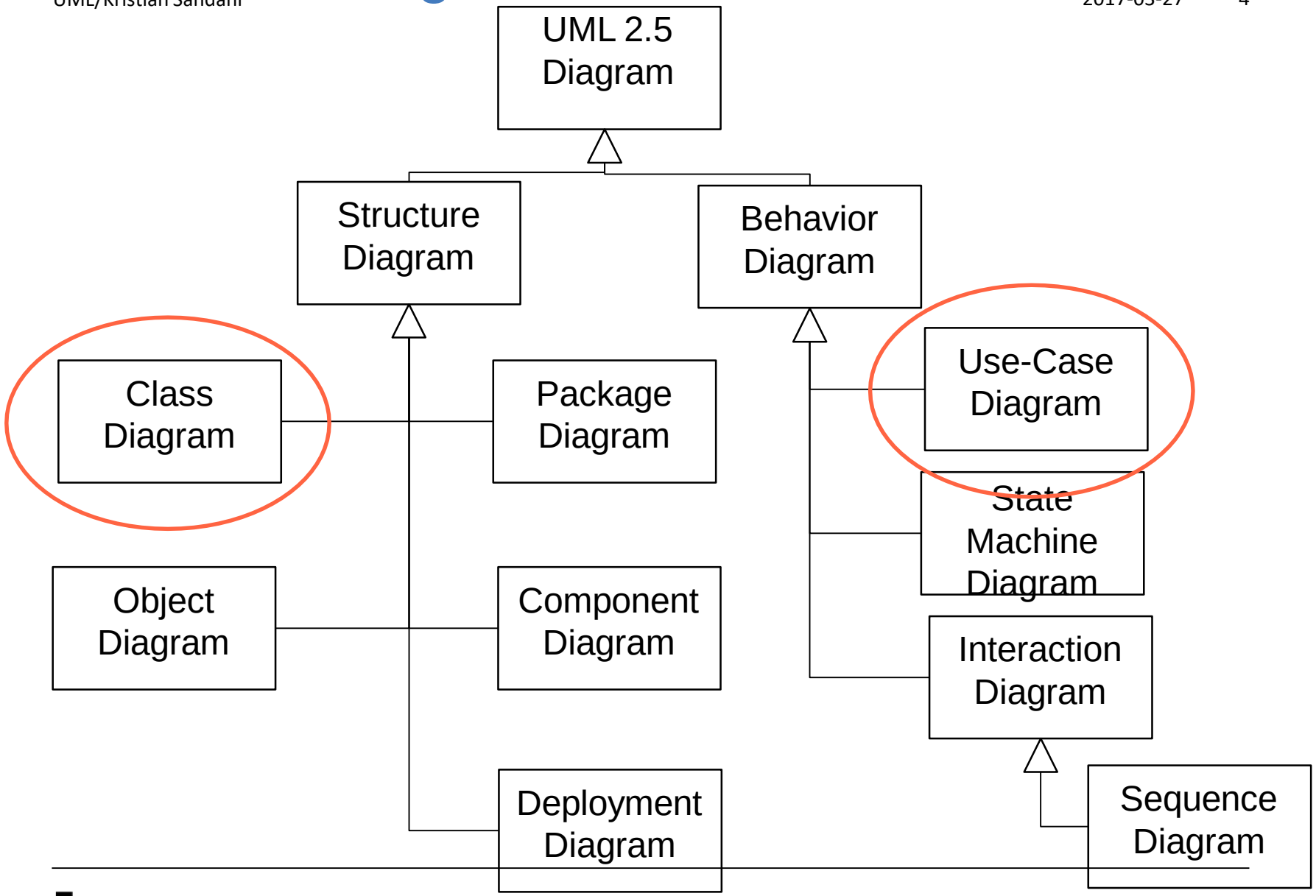


Project Management, Software Quality Assurance (SQA), Supporting Tools, Education

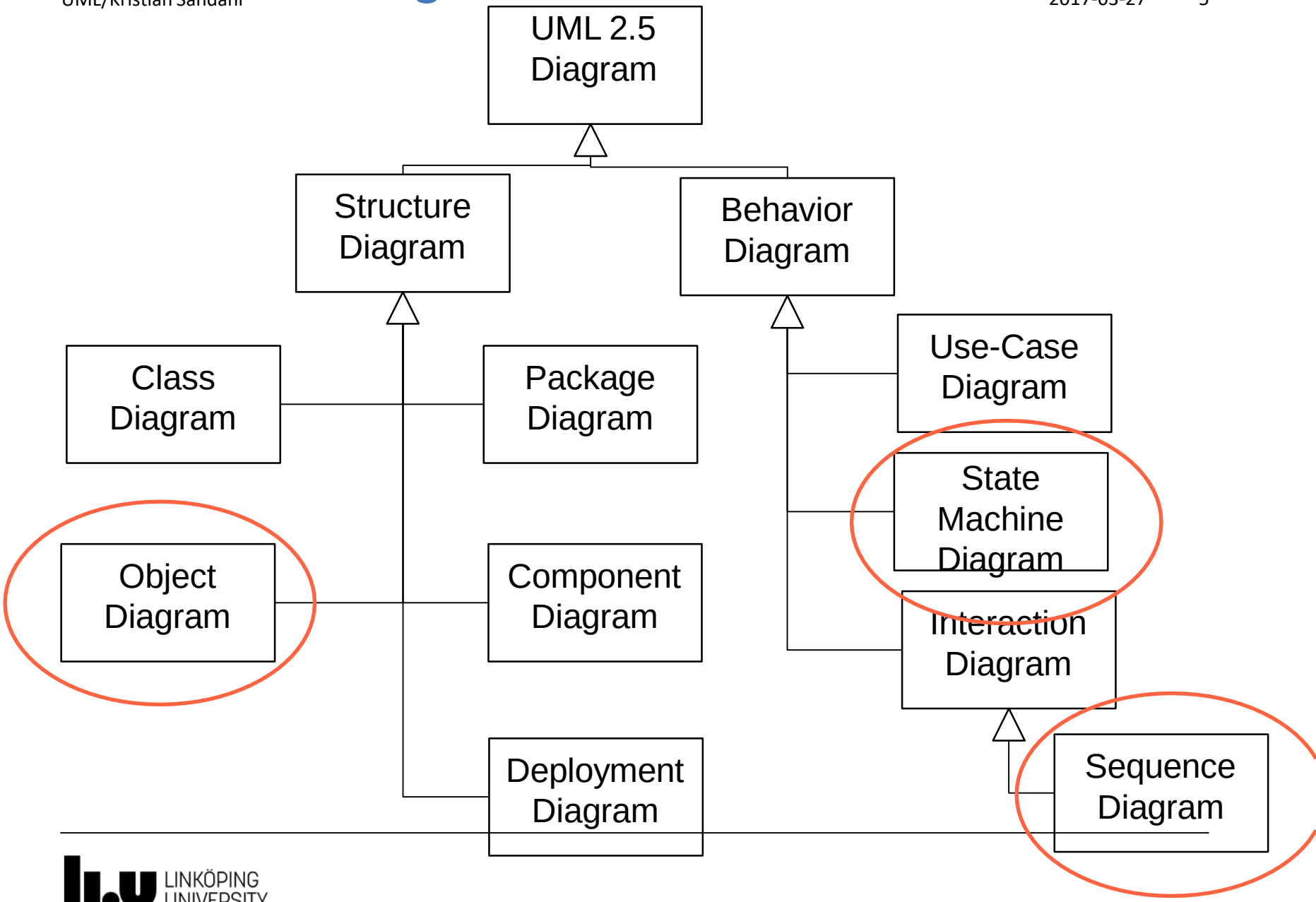
The goals of module design, again

- Provide the expected function
- Prepare for change:
  - Separation of concern
  - Testability
  - Understandability
- Contribute to quality, eg:
  - Performance
  - Usability
  - Reliability
  - ...
- Map for the implementers and testers

Well-known Diagrams of UML

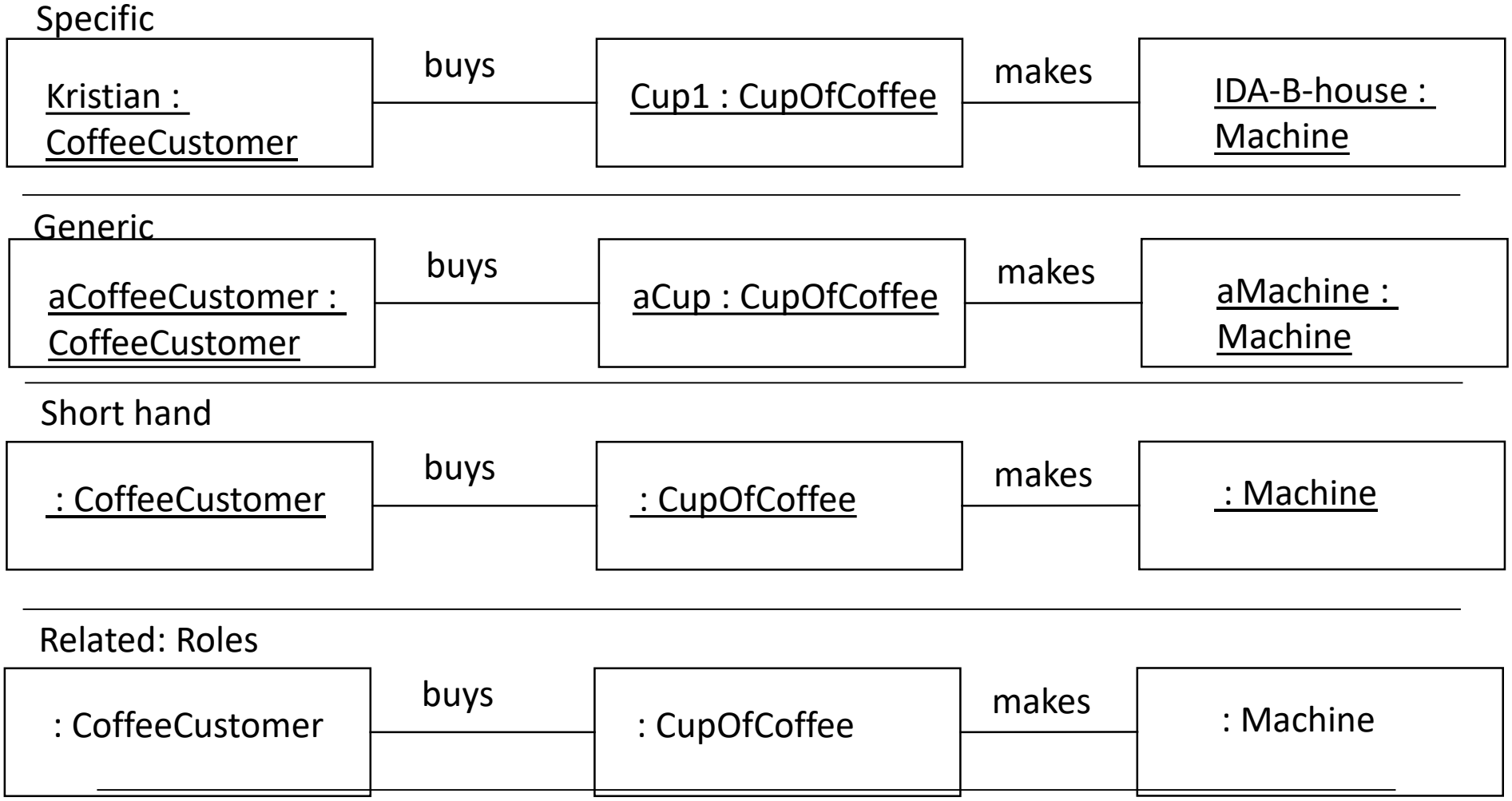


Well-known Diagrams of UML



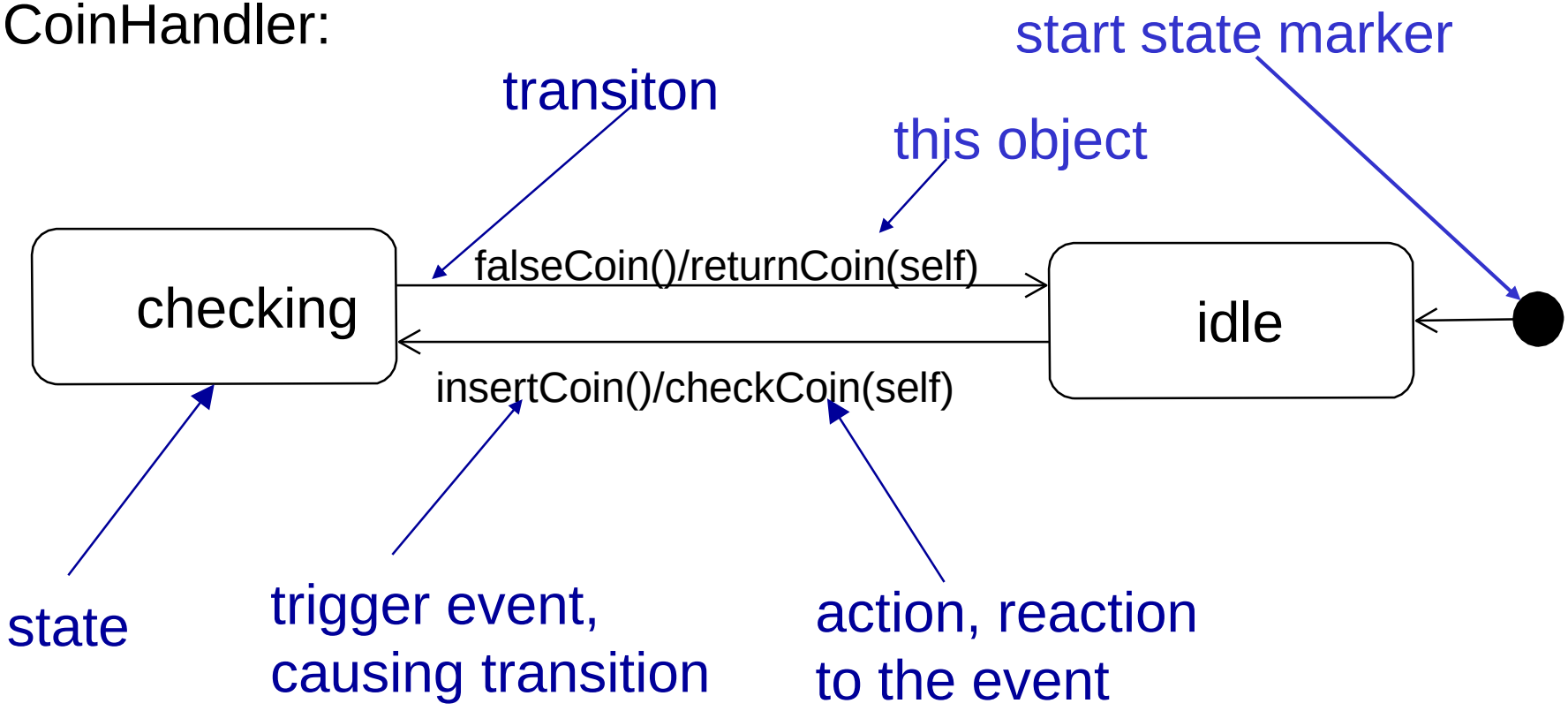
Different instance models

Course standard



State machine diagram

For class CoinHandler:

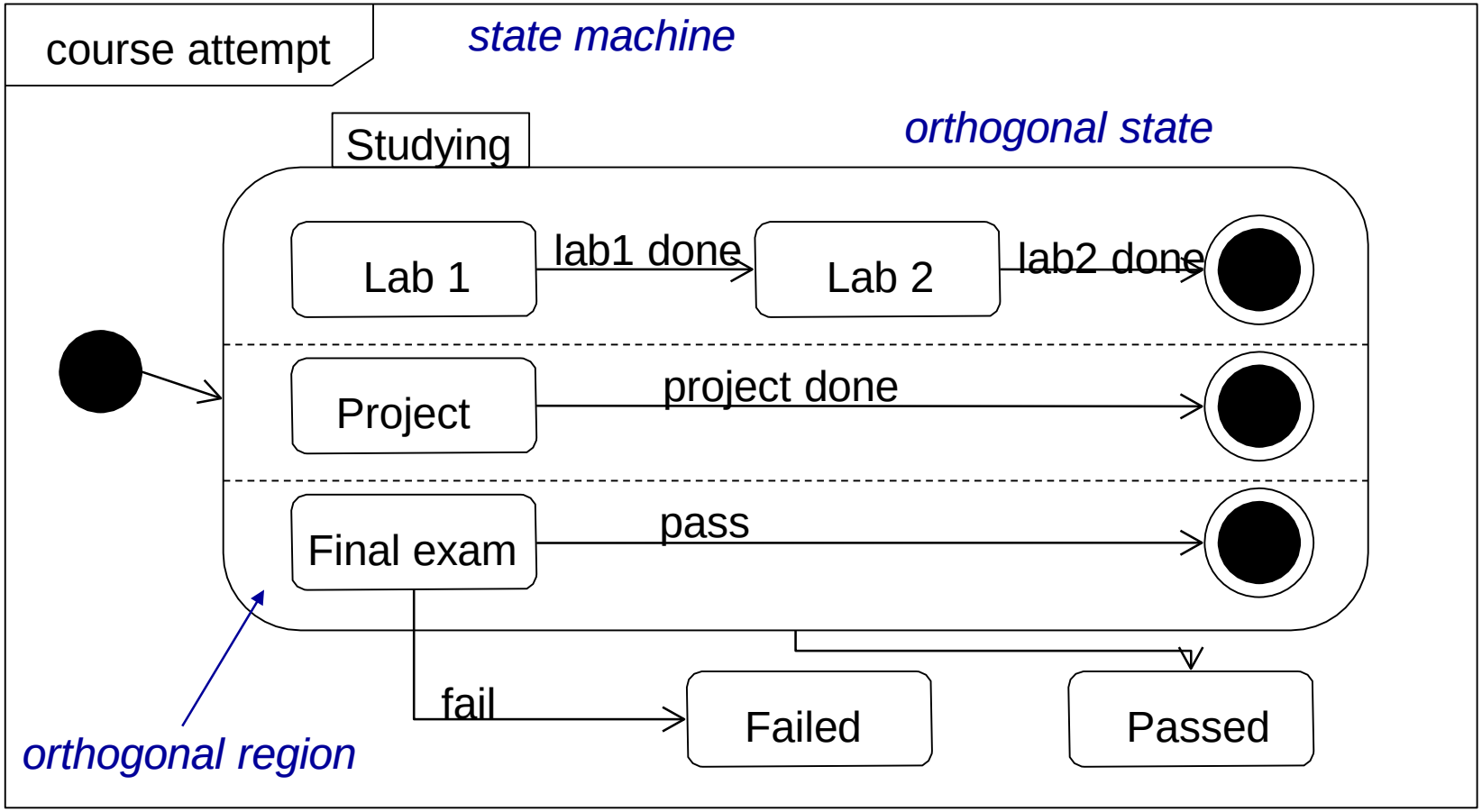


A few more states

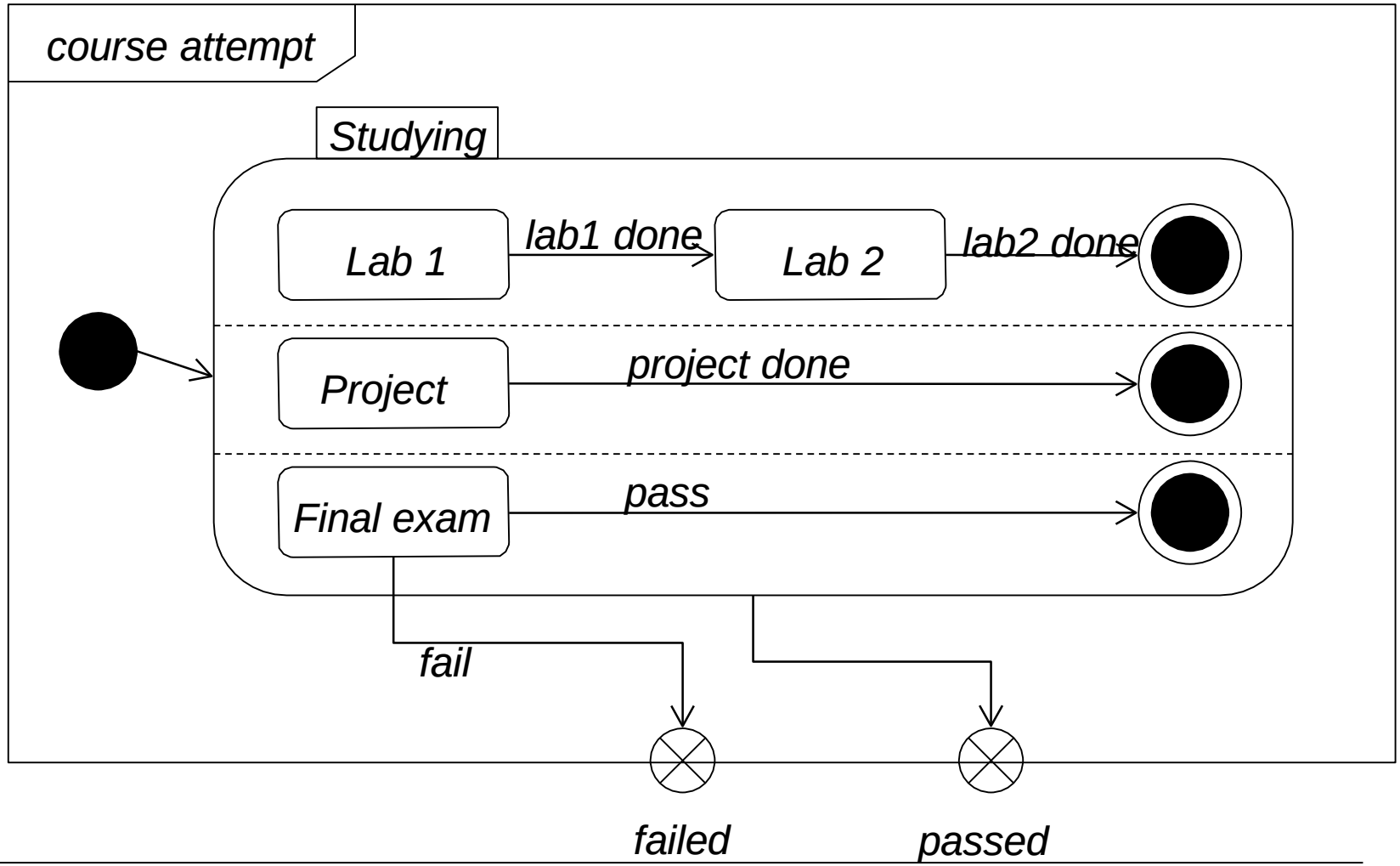
- Kristian's alarm clock starts sounding at 6.00 with a nasty signal. He can now do either of three things:
  a) Turn the alarm off;
  b) Press the snooze button; or
  c) Do nothing.

  If the snooze button is pressed the signal will turn off and start sounding after 5 minutes again.
- When an hour has passed from the first time the alarm sound started, the snooze button has no effect.
- After that the alarm sound starts, the signal will last for 2 minutes.
- If no action has been taken during these 2 minutes, the absence of action will have the same effect as if the snooze button were pressed exactly when the alarm stopped to sound.

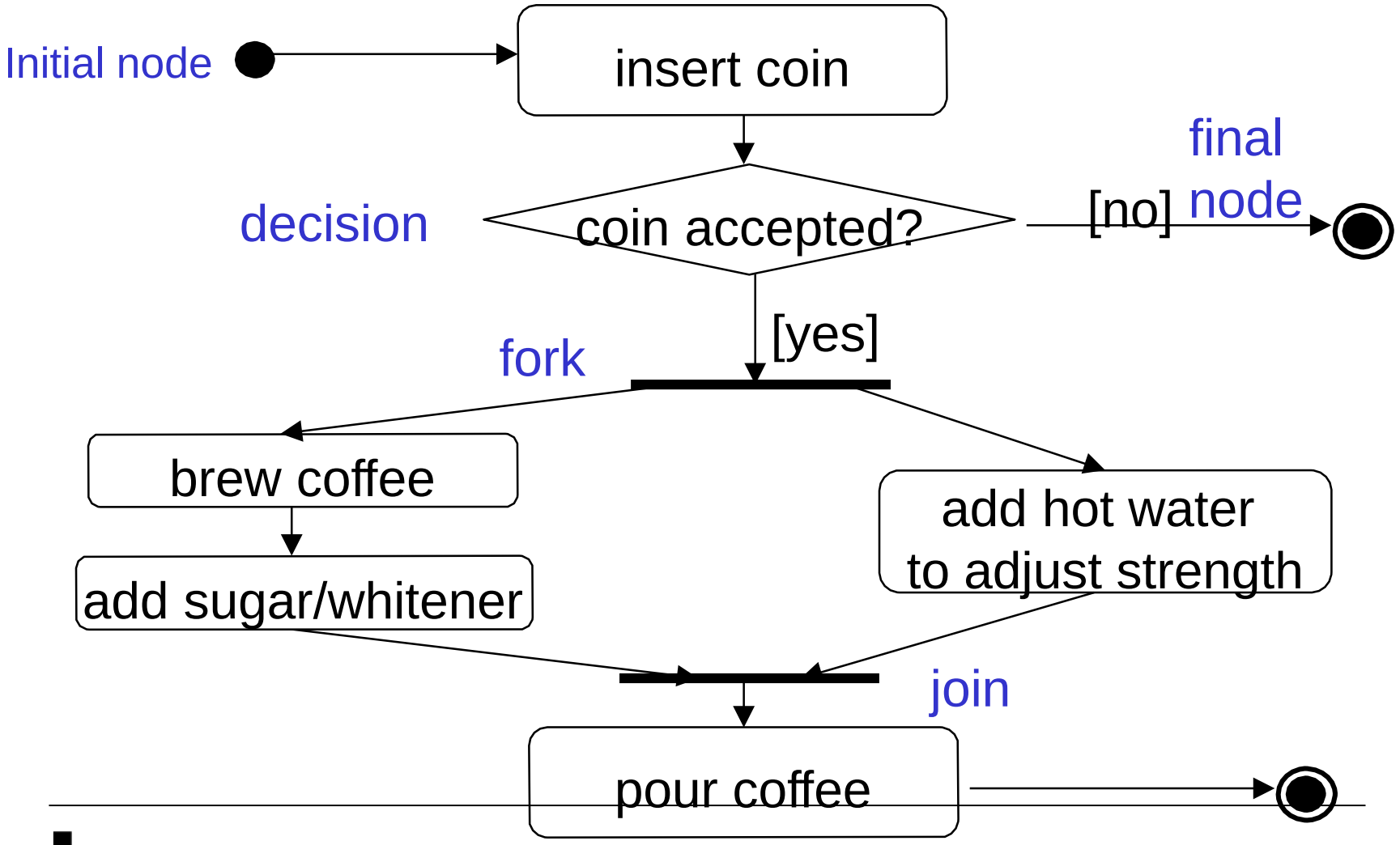
Orthogonal, composite states



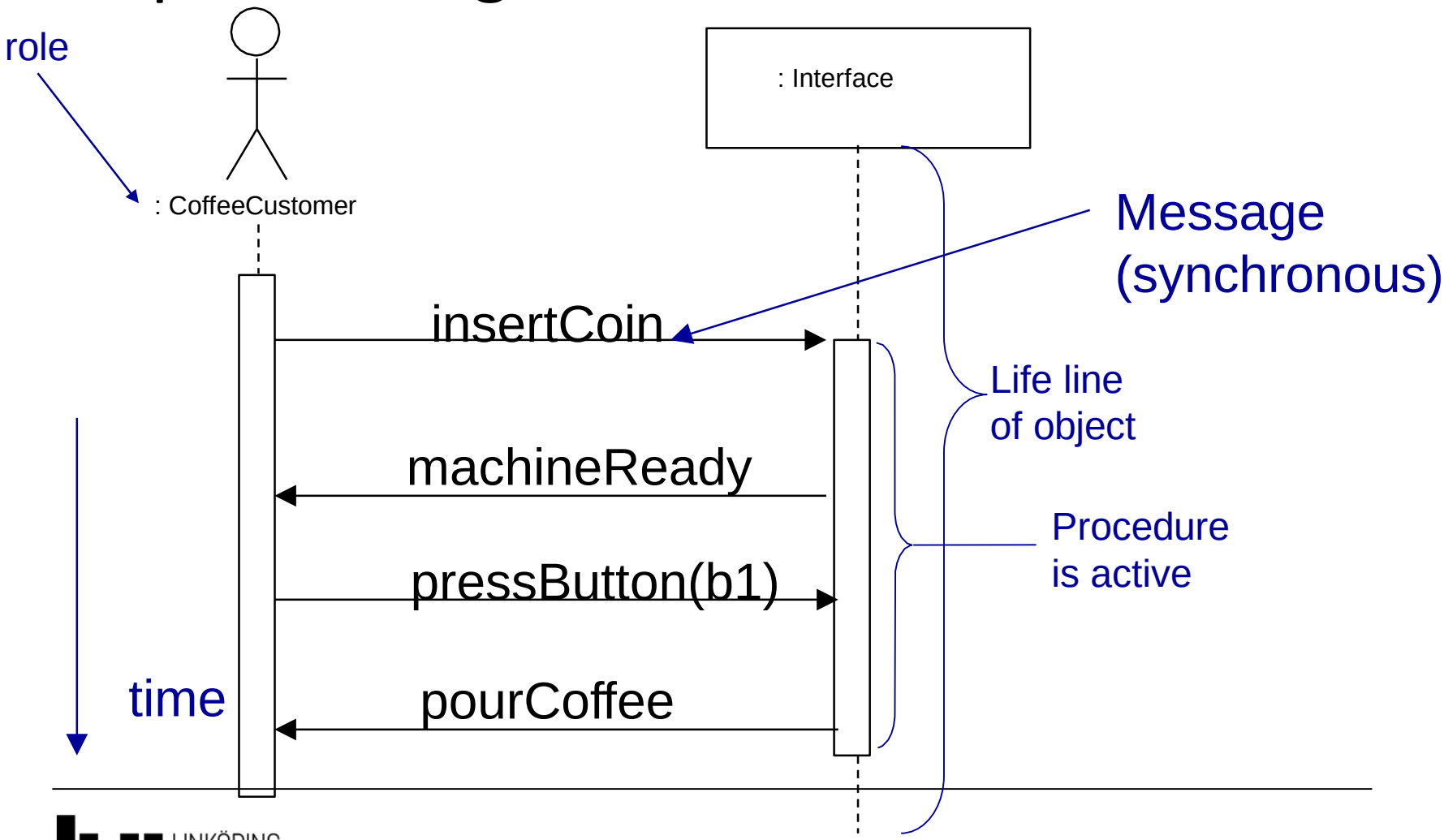
Explicit exit points



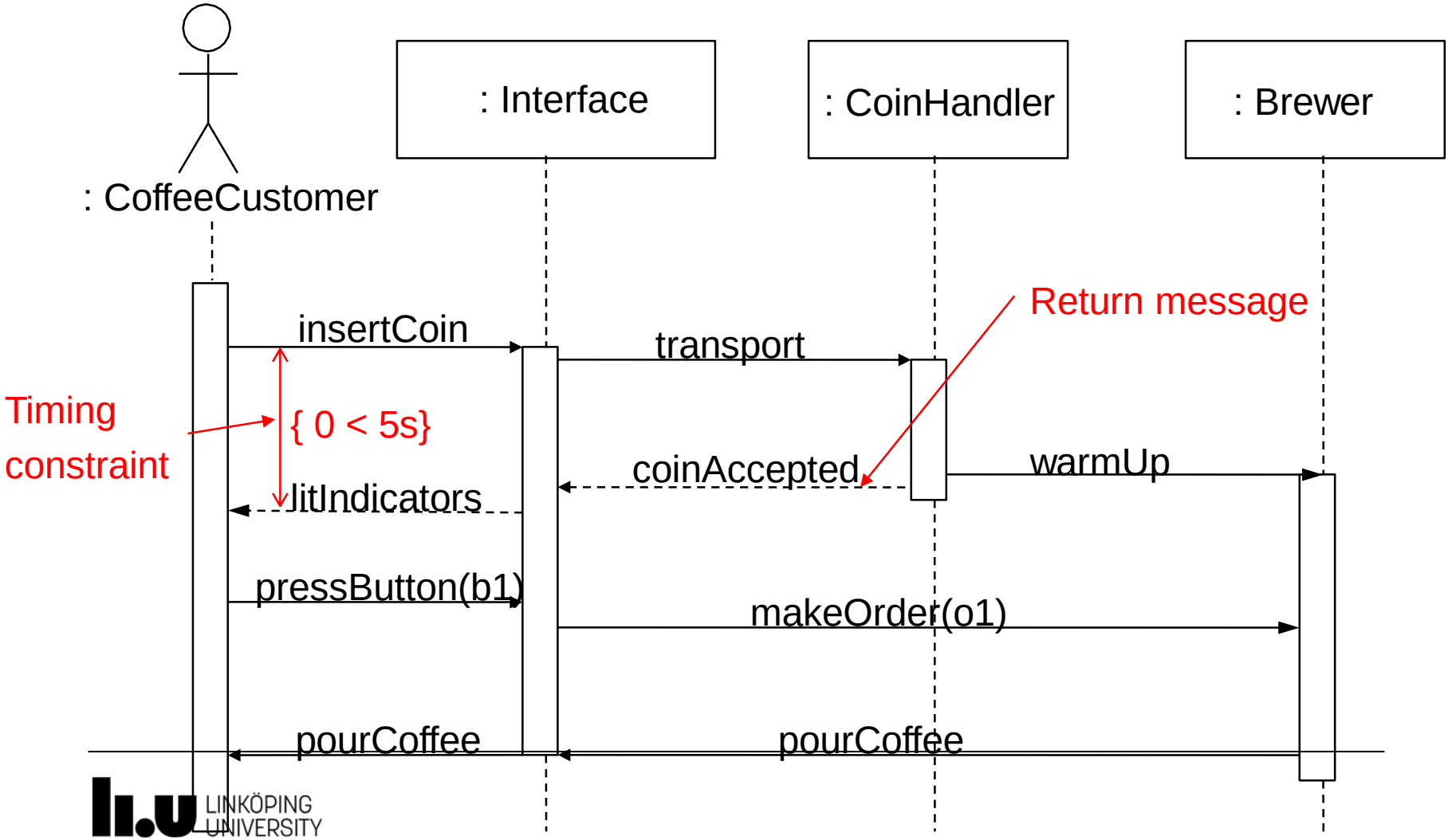
Activity diagram ≠ State diagram



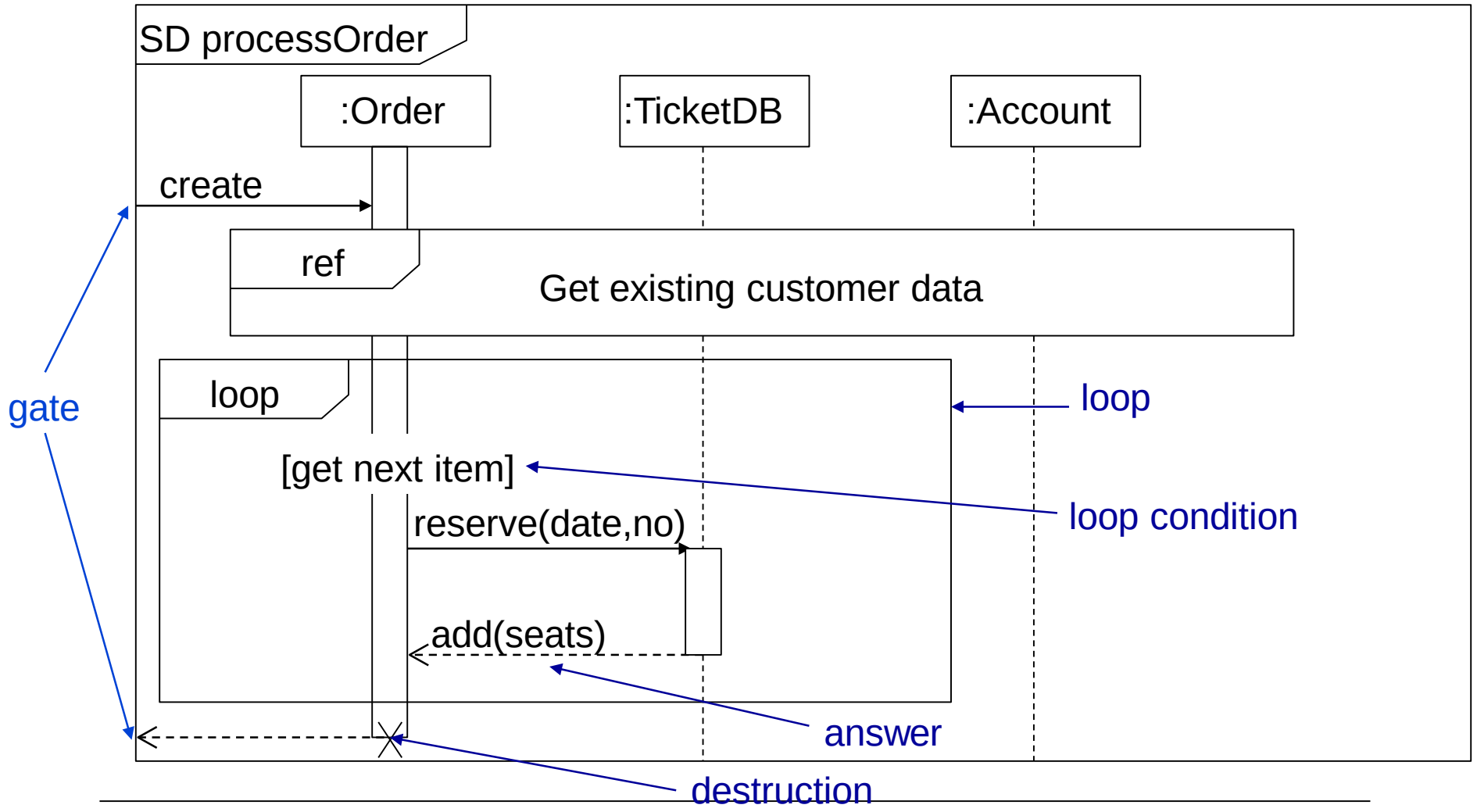
Sequence diagram



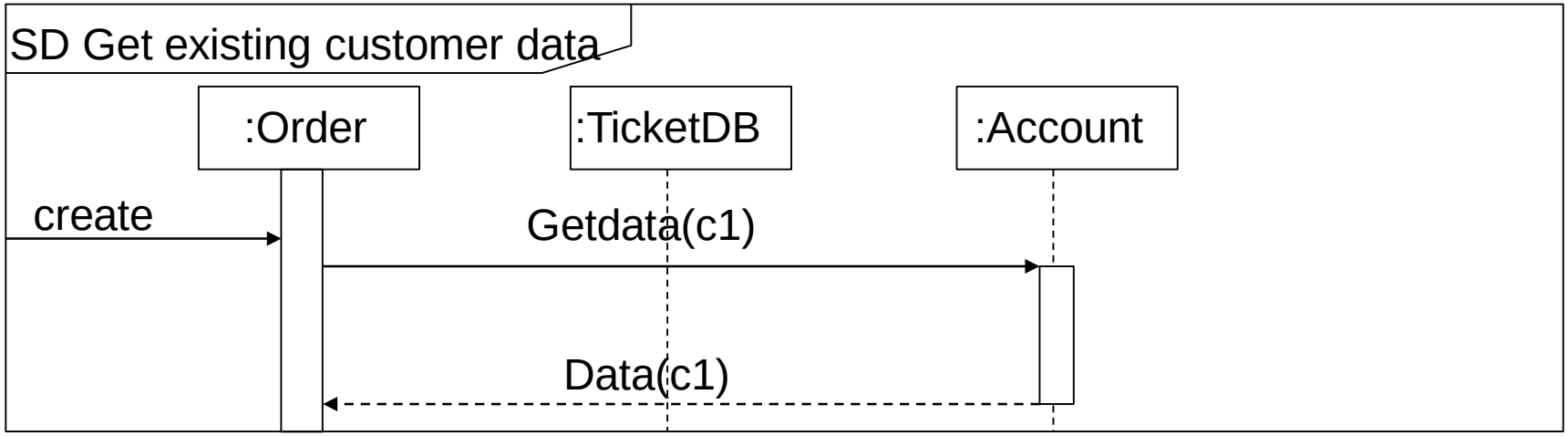
Sequence diagram with several roles



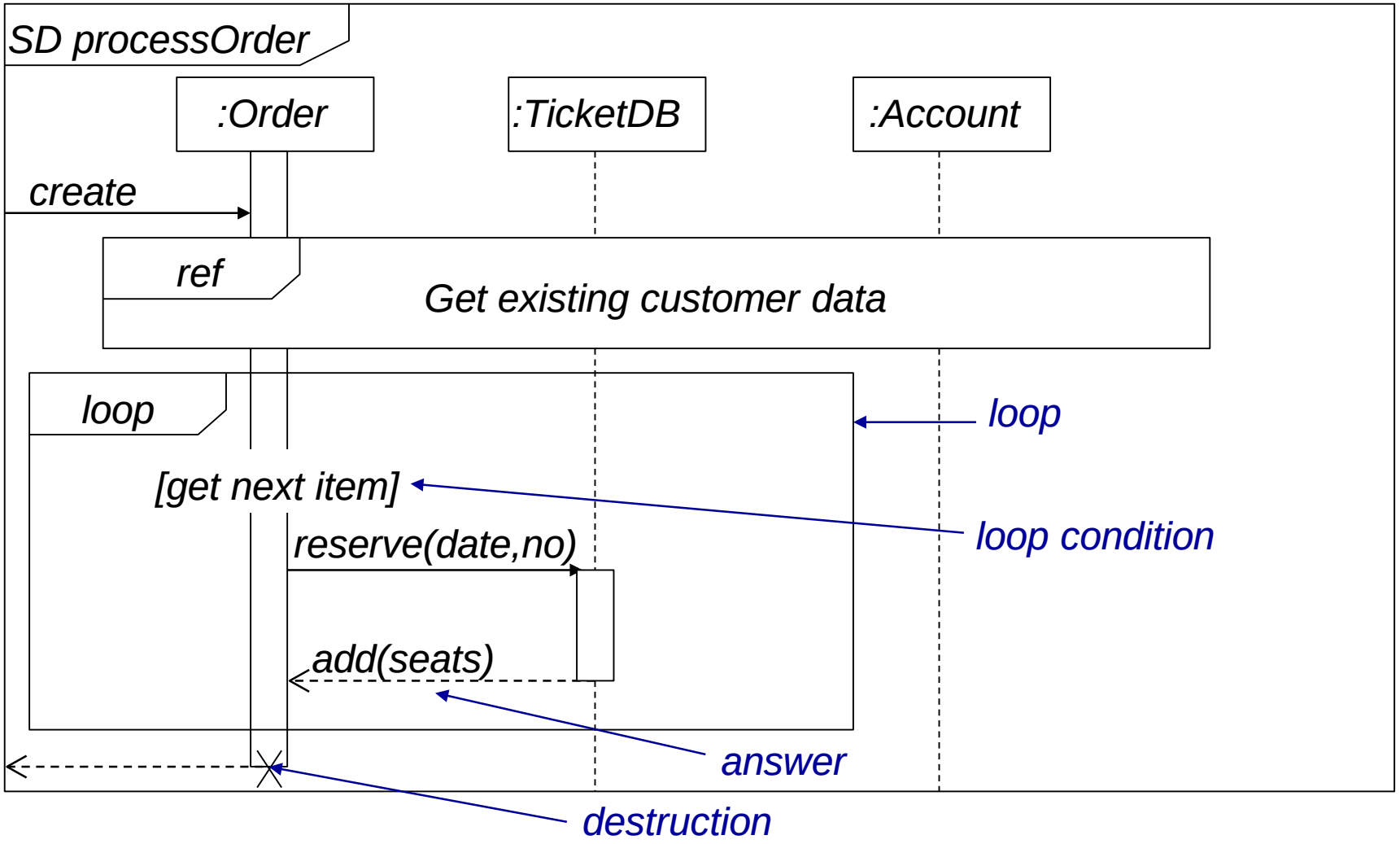
Combining fragments of sequence diagrams



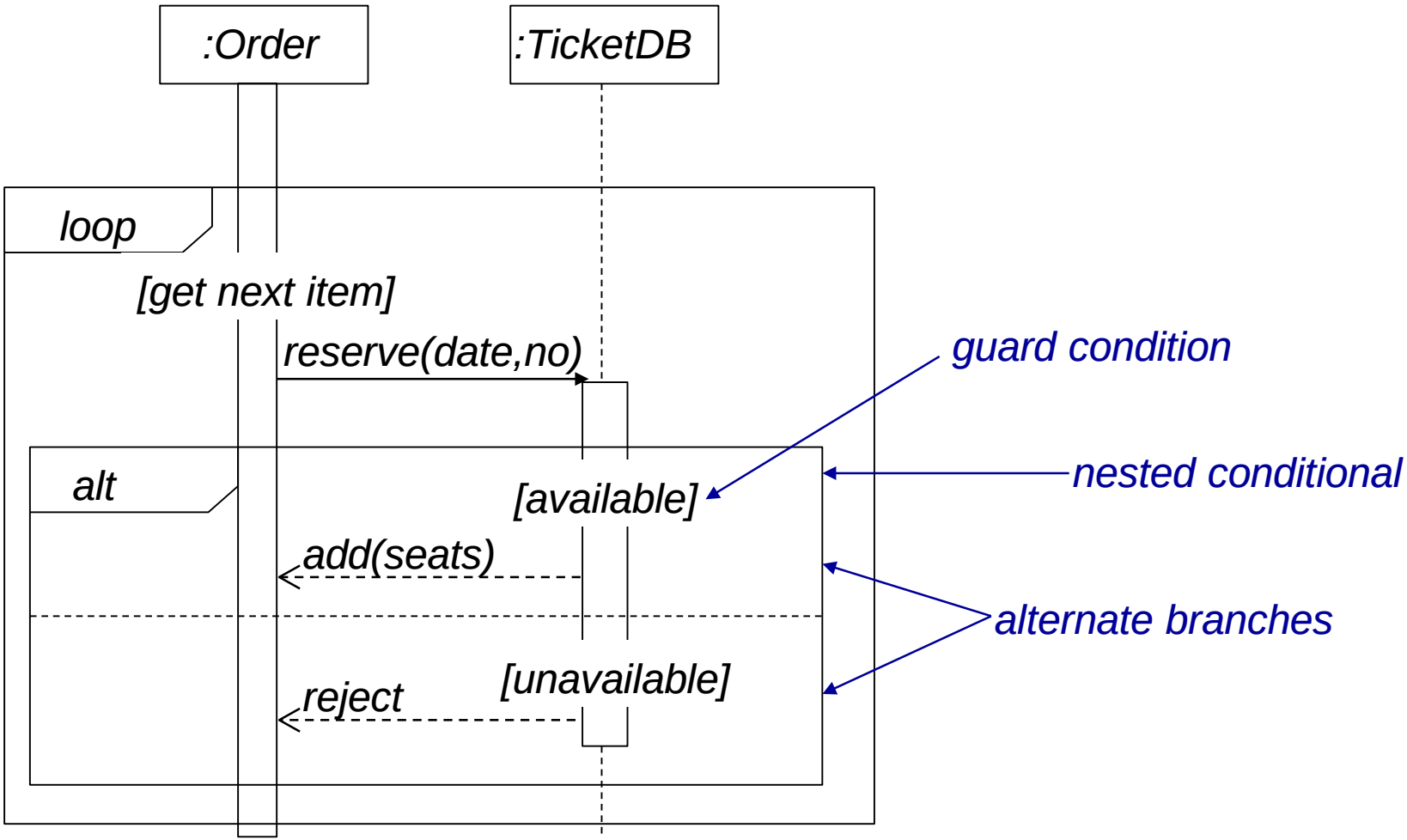
Combining fragments of sequence diagrams



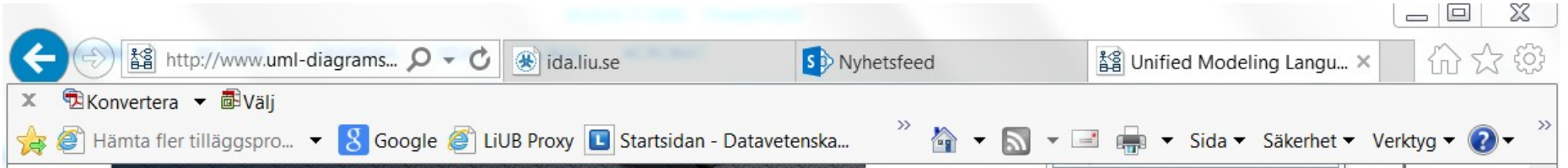
Combining fragments of sequence diagrams



More fragments of sequence diagrams



http://www.uml-diagrams.org



The Unified Modeling Language

The **Unified Modeling Language™ (UML®)** is a standard visual modeling language intended to be used for

- modeling business and similar processes,
- analysis, design, and implementation of software-based systems

UML is a common language for business analysts, software architects and developers used to describe, specify, design, and document existing or new business processes, structure and behavior of artifacts of software systems.

UML can be applied to diverse **application domains** (e.g., banking, finance, internet, aerospace, healthcare, etc.) It can be used with all major object and component **software development methods** and for various **implementation platforms** (e.g., J2EE, .NET).

UML is a standard modeling **language**, not a **software development process**. UML 1.4.2 Specification explained that process:

- provides guidance as to the order of a team's activities,
- specifies what artifacts should be developed,
- directs the tasks of individual developers and the team as a whole, and
- offers criteria for monitoring and measuring a project's products and activities.

UML is intentionally **process independent** and could be applied in the context of different processes. Still, it is most suitable for use case driven, iterative and incremental development processes. An example of such process is **Rational Unified Process** (RUP).

UML is not complete and it is not completely visual. Given some UML diagram, we can't be sure to understand depicted part or behavior of the system from the diagram alone. Some information could be intentionally omitted from the diagram, some information represented on the diagram could have different interpretations, and some concepts of UML have no graphical notation at all, so there is no way to depict those on diagrams.

For example, semantics of **multiplicity of actors** and **multiplicity of use cases** on **use case diagrams** is not defined precisely in the UML specification and could mean either concurrent or successive usage of use cases.

Name of an **abstract classifier** is shown in italics while **final classifier** has no specific graphical notation, so there is no way to determine whether classifier

UML/Kristian Sandahl

www.liu.se

LINKÖPING UNIVERSITY