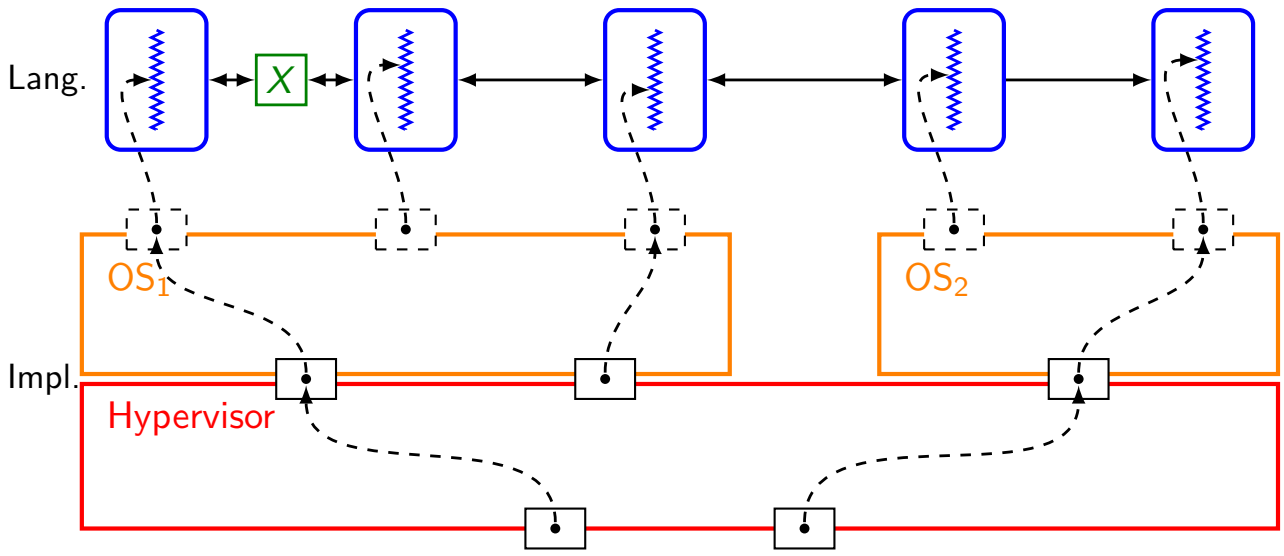
Course 02158

Basic Concepts

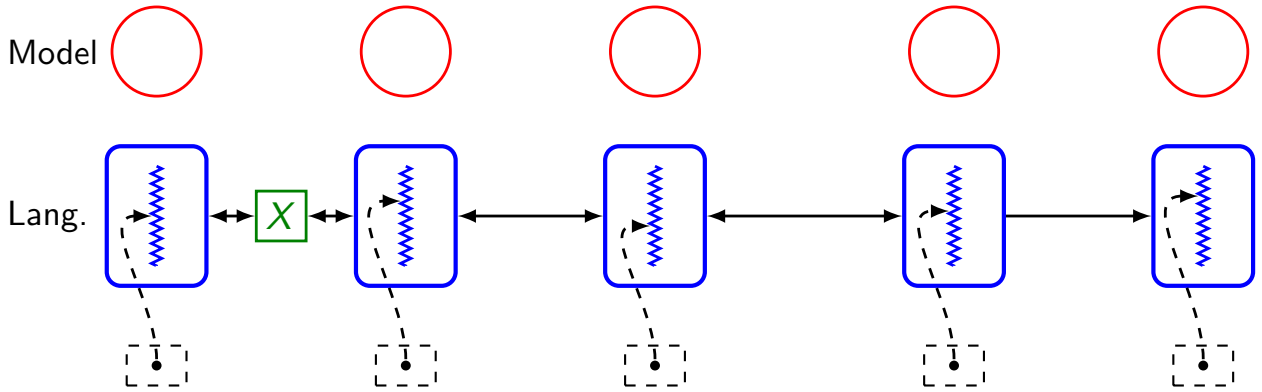
Hans Henrik Løvengreen

DTU Compute

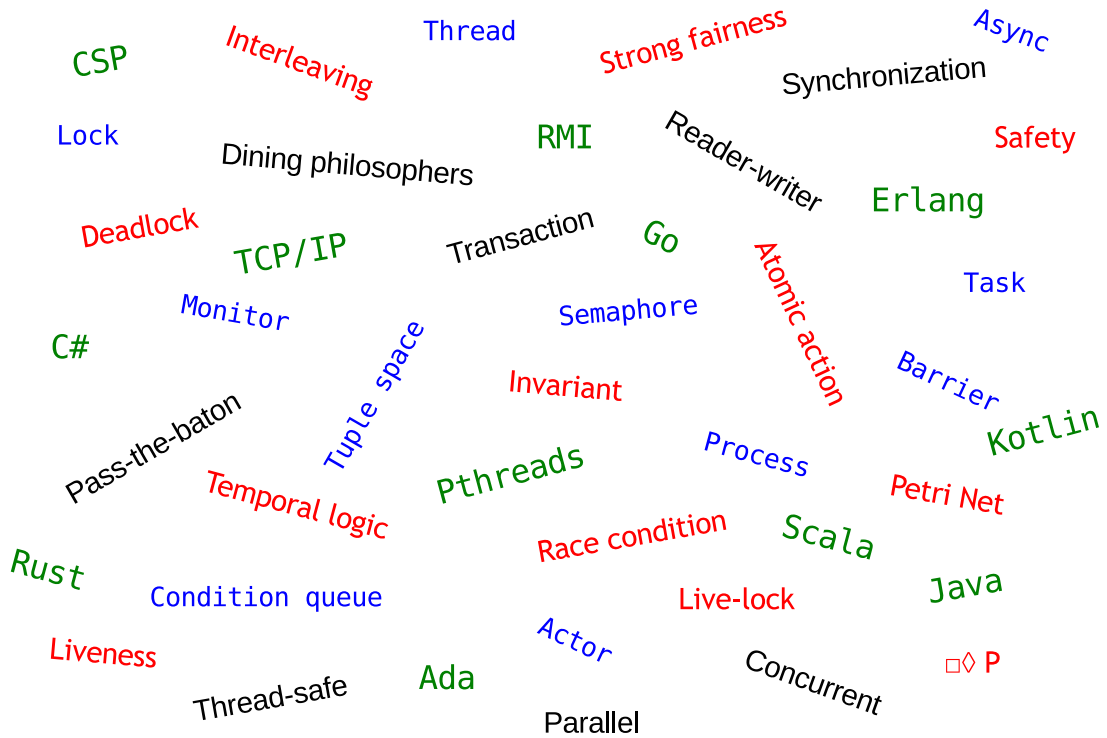
Concurrent Systems



Concurrent Programming



Concepts and Notions



Course Domain

Systems of

synchronized, *concurrent*, *sequential processes*

Processes

- Generally characterized by:
  - Being *activities* over (long) time
  - Involving *gradual change of state*
  - Being *regular*

Abstract Process Notion

- Assumption: State changed by discrete *actions*
- *Process* = behaviour of part of system (given by action set)
- *Behaviour* = set of potential executions
- *Execution* = observation of action occurrences

NB

- OS Process = program execution context

Concurrent Processes

- Two actions are *concurrent* if they may be executed in parallel
- *Parallel execution* = overlapping in time
- A *sequential process* has no concurrent actions
- *Processes are concurrent* if they have concurrent actions

Synchronization

- Synchronization = constraint on the ordering of actions

Condition Synchronization

- "Data must be ready before being read"
- Reflects *causality*

Mutual Exclusion

- "At most one process may use the printer at a time"
- Reflects *resource sharing*

True Synchronization

- "All participants meet before each round"
- Also known as *barrier synchronization*
- Reflects *cooperation*