Real Time Systems – SS2016

Prof. Dr. Karsten Weronek

Faculty 2

**Computer Science and Engineering**

Dependent tasks

Dependent tasks

The assumption of independent tasks often does not meet reality.

- In practice, tasks are dependent.
  We often have conditions or constraints e.g.

  - A must be computed before B

  - B must be computed before C and D

- Such conditions are called **precedence constraints** which can be represented as Directed Acyclic Graphs (DAG) known as **Precedence Graphs**

- Such graphs are also known as „**Task Graph**"

- ## Input/output relation
  - ### Some task is waiting for output of the others: data flow diagrams
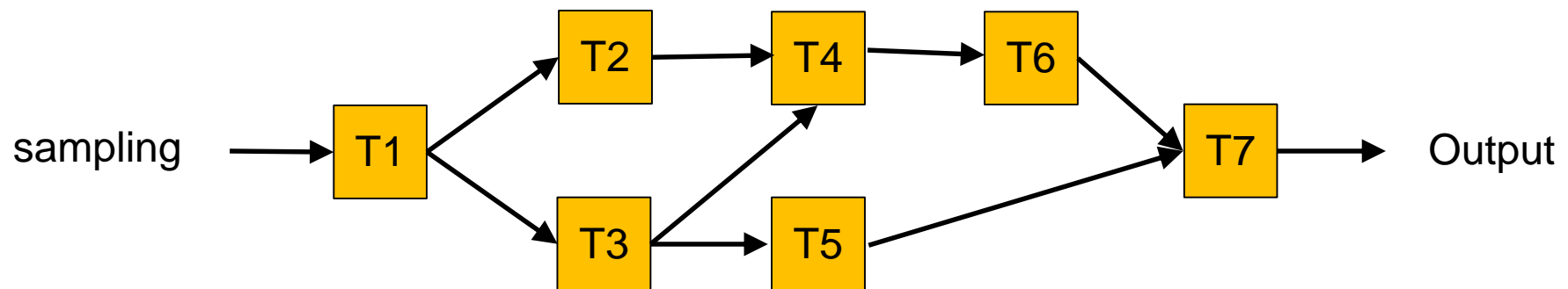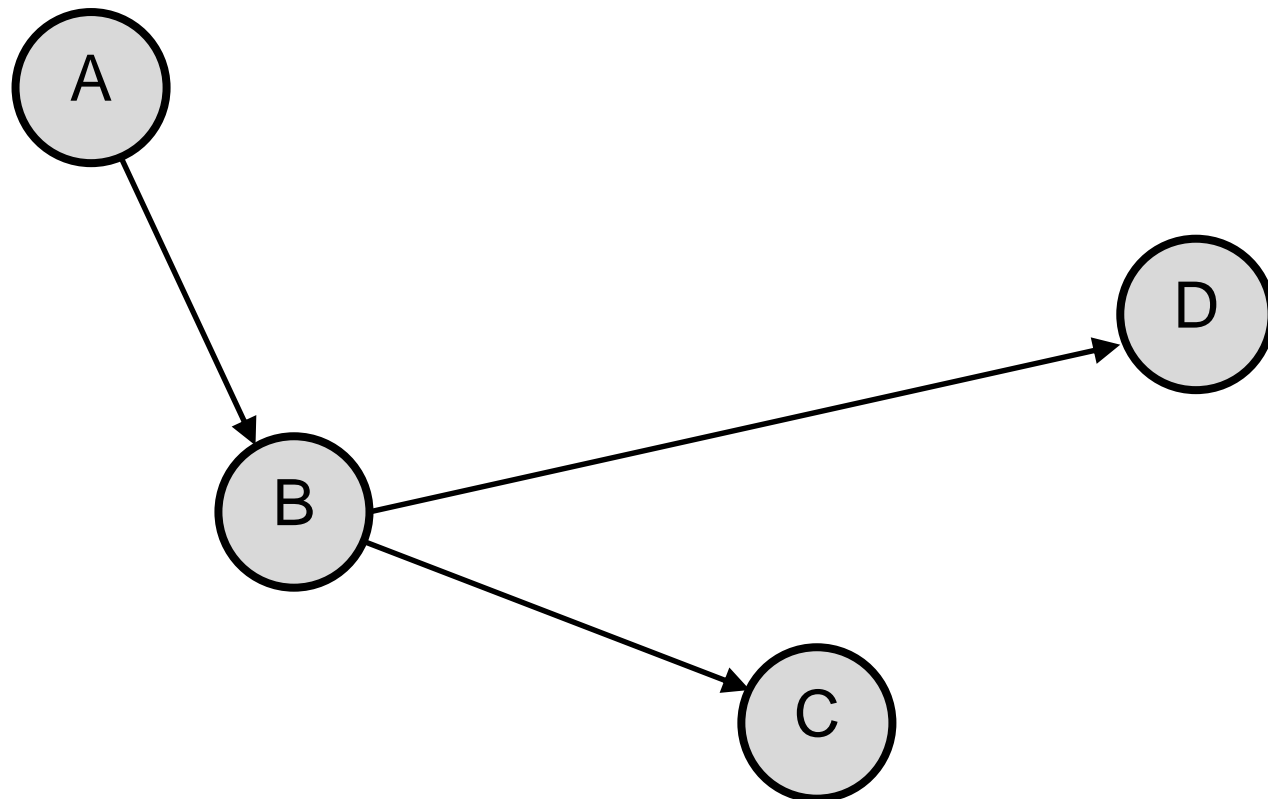

- ## Synchronisation
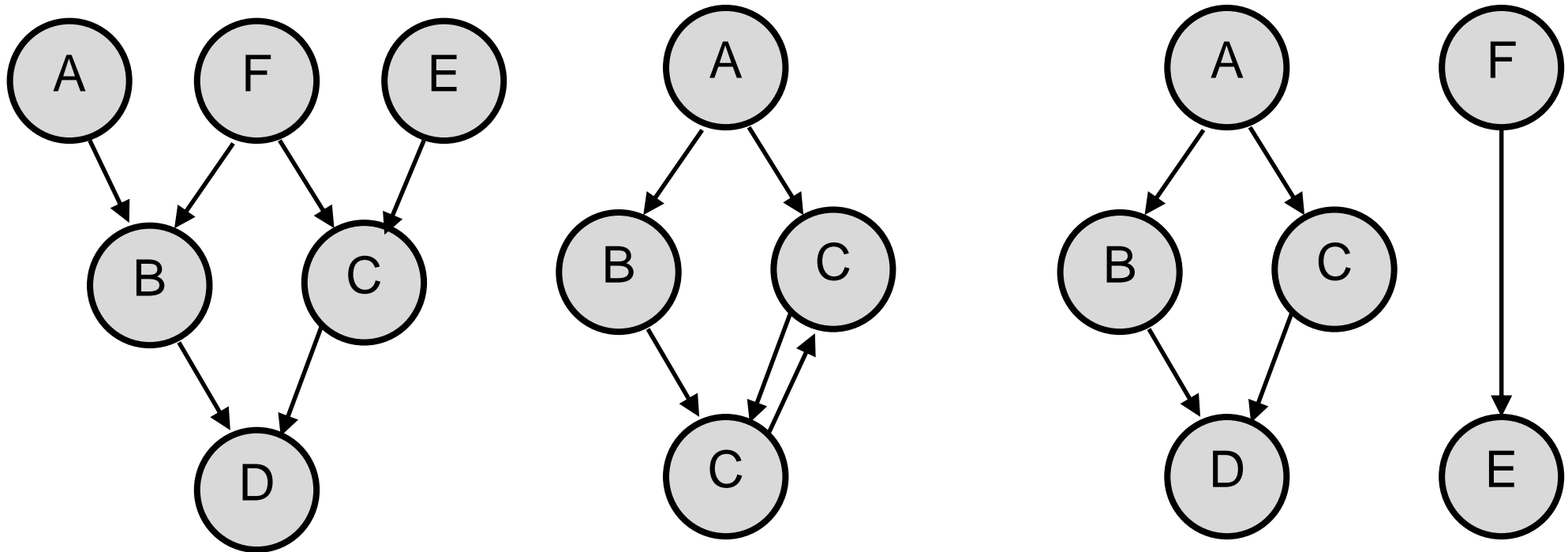 - ### Some task must be finished before the others e.g. It is holding a shared resource

- ## Other dependence relations (Resource lock, shared resources, etc.)

- A must be computed before B
- B must be computed before C and D

Nodes(Knoten) and edges(Kanten)

Not a precedence graph

Conjunct and dishunct join (not considered)

- AND-node, all incoming edges must be finished first
- OR-node: at least one of the incoming edges must be finished

Scheduling under timing and precedence constraints

- **Feasible schedules should meet**
  - **Timing constraints**
  - **Precedence constraints**

- **Overlapping area of blue and red is what we need**
- **Precedence constraints restrict the search area.**

Schedules satisfying precedence constraints

Schedules satisfying timing constraints

All possible schedules (feasible/unfeasible)