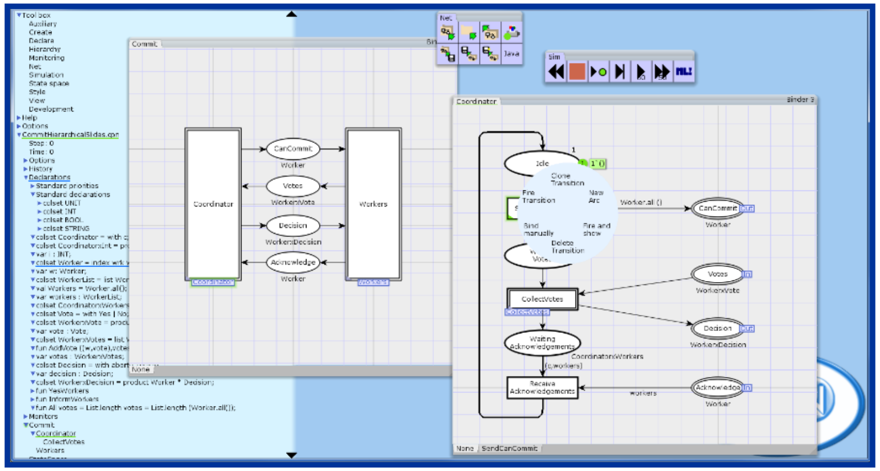
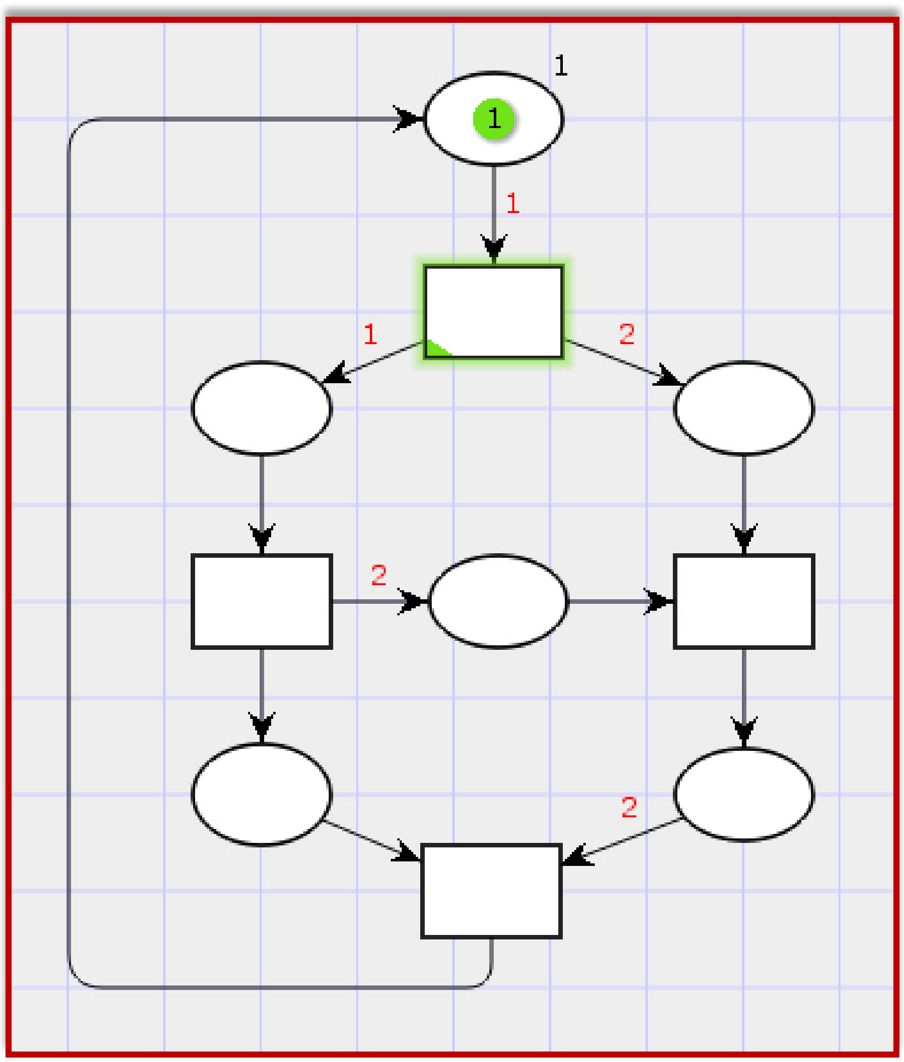
Lecture 2

Two-phase Commit Protocol and Place/Transition Nets



Lars M. Kristensen
Department of Computing, Mathematics, and Physics
Western Norway University of Applied Sciences
Email: lmkr@hvl.no / WWW: home.hib.no/ansatte/lmkr

Quick Recap: Petri Net Concepts



State modelling

- Places (ellipses) that may hold tokens
- Marking (state): distribution of tokens on the places
- Initial marking: initial state

Event (action) modelling

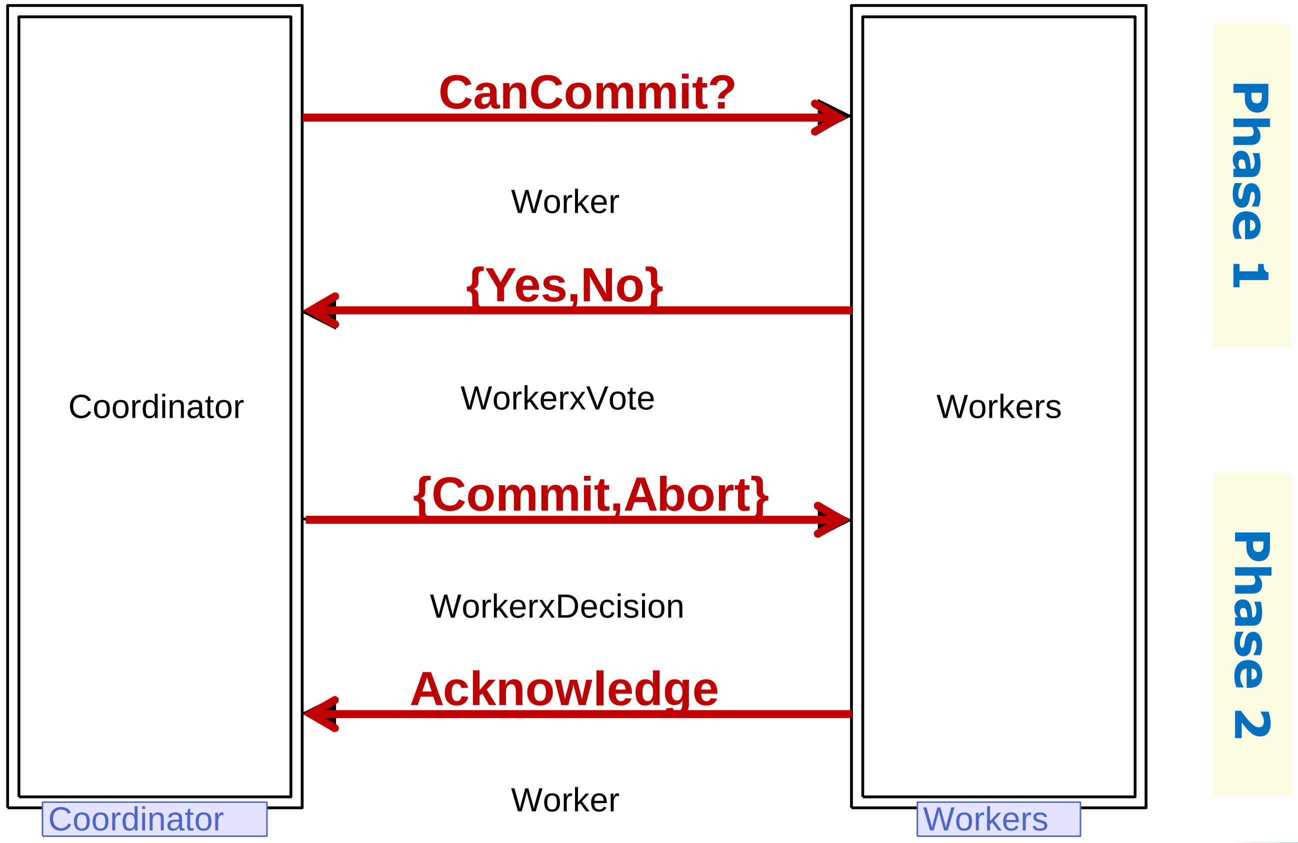
- Transitions (rectangles)
- Directed arcs: connecting places and transitions
- Arc weights: specifying tokens to be added/removed

Execution (token game)

- Current marking
- Transition enabling
- Transition ocurrence

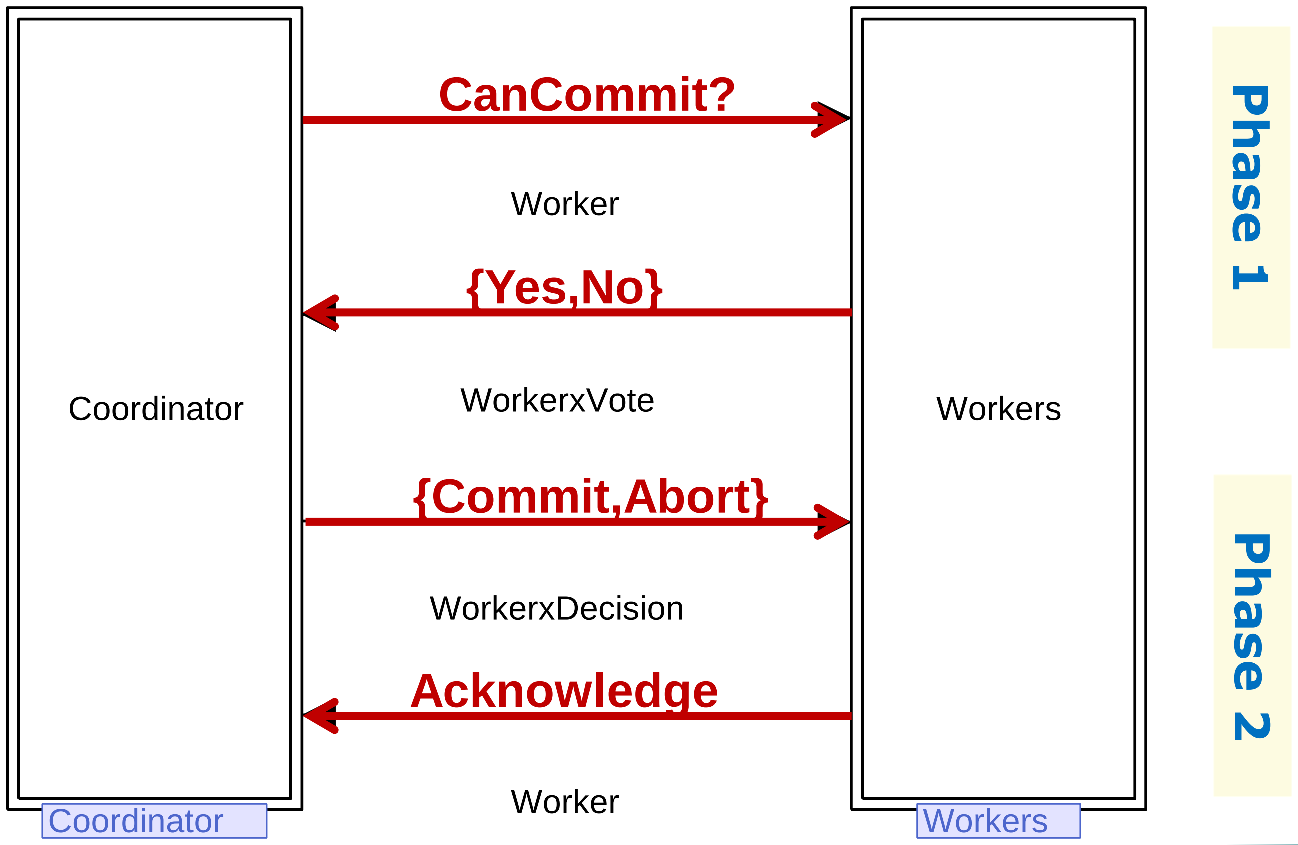
Two-phase Commit Transaction Protocol

- A concurrent system consisting of a coordinator process and a number of worker processes



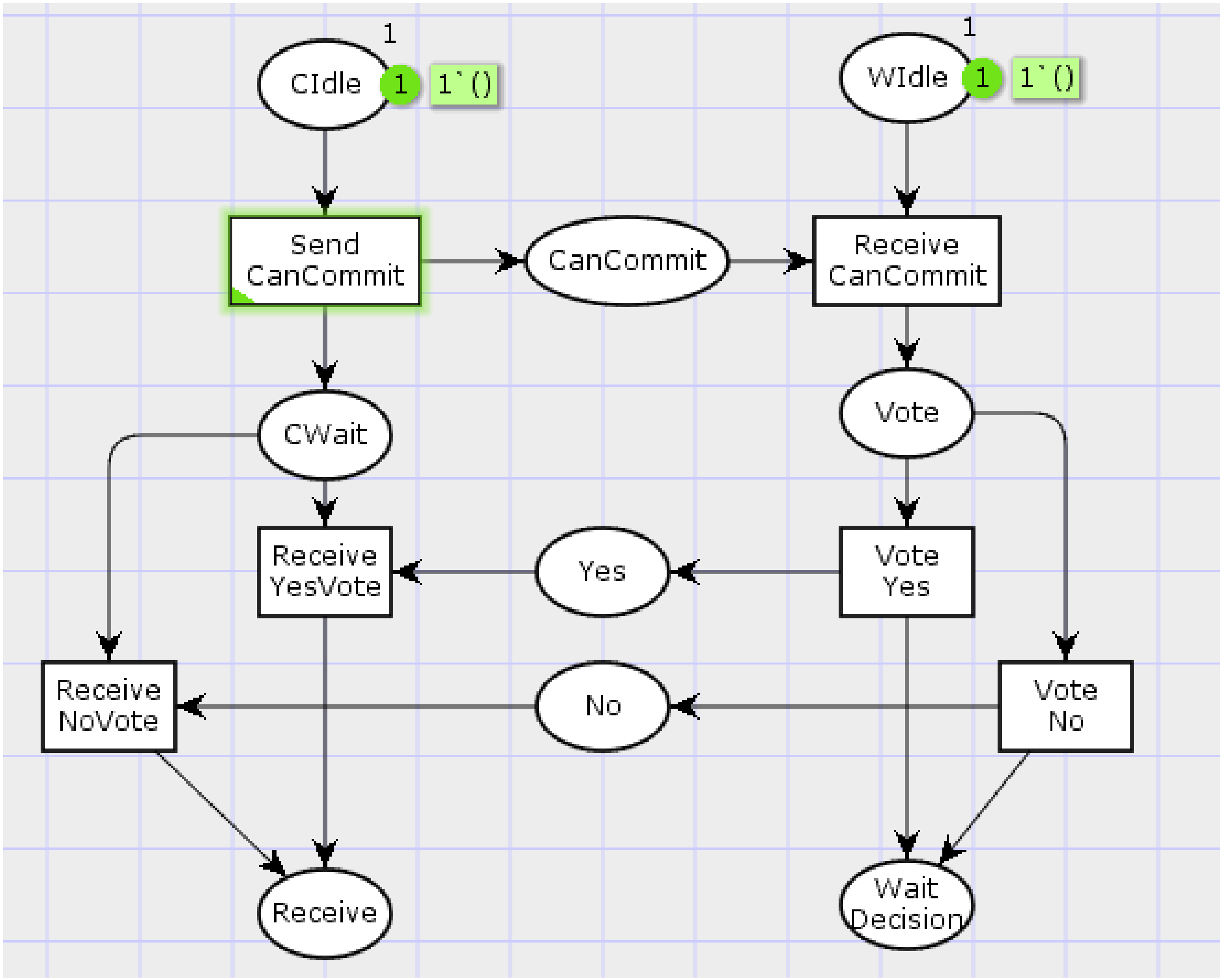
Two-phase Commit Transaction Protocol

- **How to model phase 1 with PT-nets?**



CPN Tools Demo

- **Construction, editing and simulation of basic Petri Net models**
- **First part of the two-phase commit protocol using Place/Transition Nets**
  - How to model send and receive CanCommit with one worker?
  - How to model Yes/No votes?
  - How to model multiple workers?


1
CIdle
1
1`()
1
WIdle
1
1`()
Send CanCommit
CanCommit
Receive CanCommit
CWait
Vote
Receive YesVote
Yes
Vote Yes
Receive NoVote
No
Vote No
Receive
Wait Decision

Why do we need CPNs ?

- **CPNs include the basic syntactical and semantical concepts of Place/Transition Nets**
  - The black/anonymous PT-net tokens are represented using the UNIT type and the unit value ()
- **A main limitation of Place/Transitions Nets is scalability to large (real) software systems**
  - Does not support parametric systems in an elegant way
  - Modelling of data is inconvenient
  - Does not allow models to be split into modules
- **CPNs provides additional language constructs**
  - Inhibitor arcs and reset arcs
  - Transition priorities