Process Oriented Data Science

UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

Campus d'Excel·lència Internacional

Josep Carmona
Computer Science Department

Outline

- **M1: Process Mining Overview, Positioning & Preliminaries (Event data & Process Models)**
- M2: Process Discovery
- M3: Conformance Checking
- M4: Process Enhancement

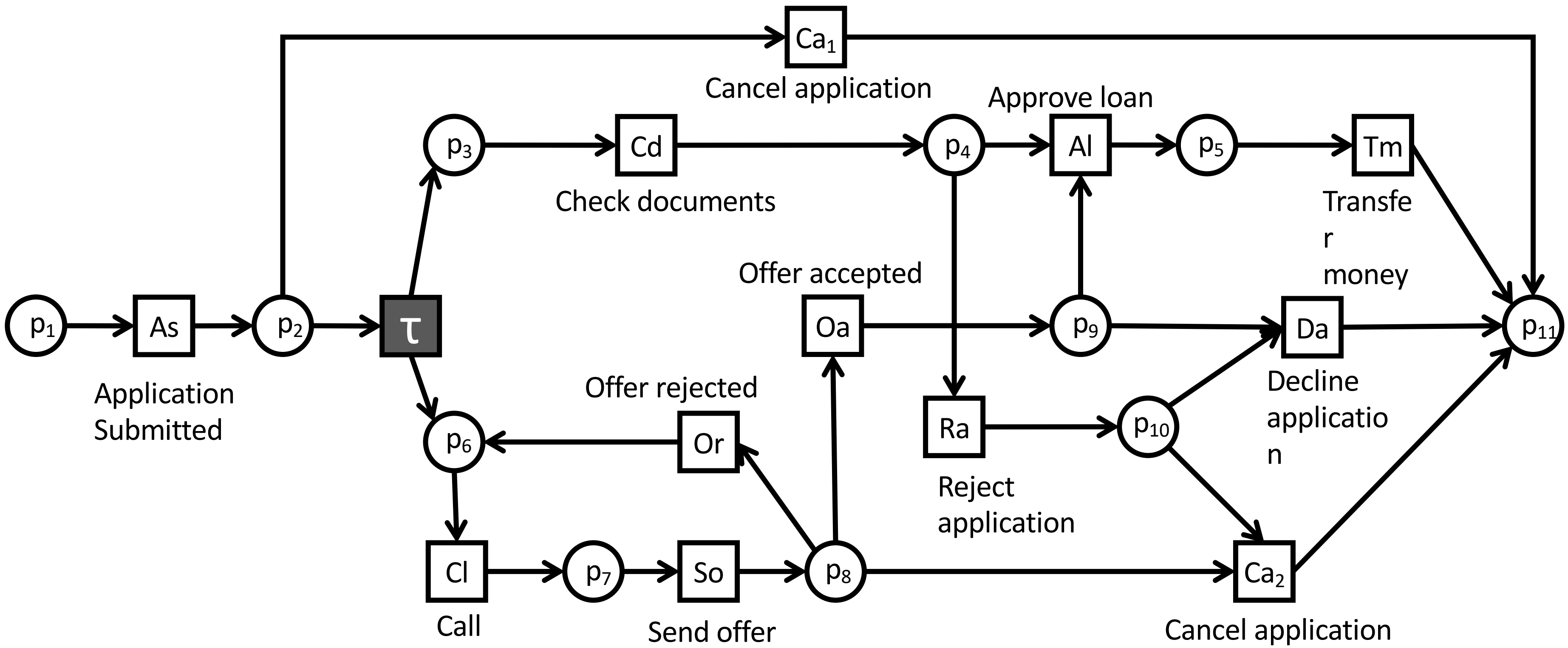
Disclaimer

- Most of the material of this course is taken from my colleagues:
  - RWTH Aachen (Prof. Wil van der Aalst)
  - Humboldt University zu Berlin (Prof. Matthias Weidlich)
  - **Technishe Universiteit Eindhoven (Prof. Boudewijn van Dongen)**
  - University of Tartu (Prof. Marlon Dumas)
  - University of Melbourne (Prof. Marcello La Rosa)
  - Technical University of Denmark (Prof. Andrea Burattin)
- Hence, this material is only provided for your learning, please do not share nor publish

Outline

- Soundness
- Petri net properties
- Special classes of Petri nets

Recall our example



Soundness of Workflow Nets

- Workflow nets (or process models) are Petri nets,
  - with a single source place i *(p_0 in our example)*,
  - with a single final place f *(p_{11} in our example)*,
  - where every node is on a path from i to f.

- A process model is sound if and only if:
  - For every marking reachable from $[i]$, you can reach the marking $[f]$
  - No marking *m* can be reached with $m > [f]$
  - There are no dead transitions (for every transition t, a marking m can be reached from $[i]$ such that m enables t, i.e. $m \geq \bullet t$)

Importance of Soundness

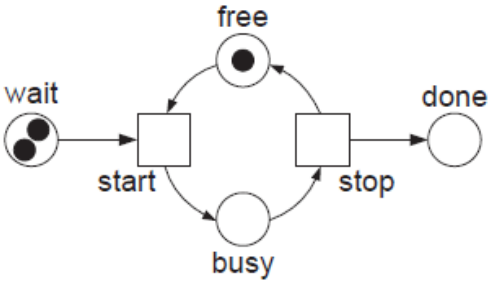
- Unsound models may get stuck (deadlocks)
- Unsound models may never be able to terminate (livelocks)
- Unsound models cannot be simulated / analyzed
- Unsound models cause problems in the processes they model!

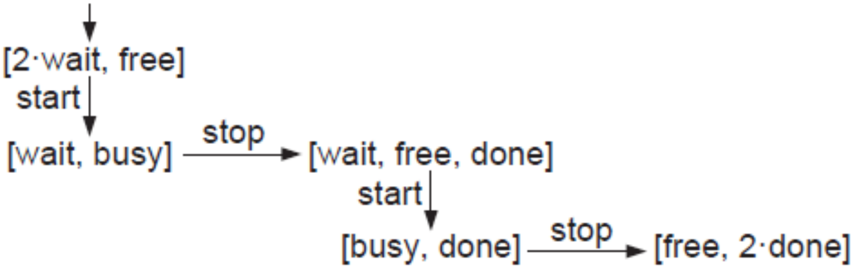
Petri net Properties

- Reachability
- Reversibility
- Boundedness
- Liveness
- Deadlock(freedom)

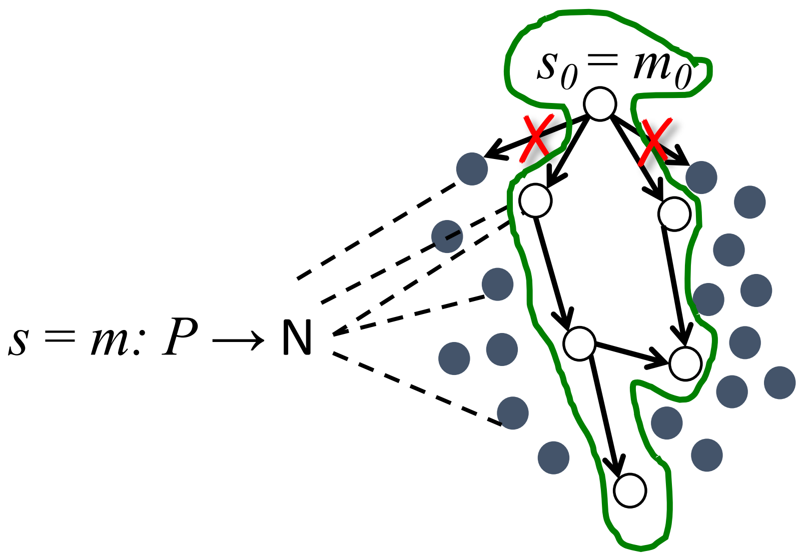
Reachability graph

- Recall mapping of a Petri net onto a transition system:
  - Transition-system states: markings reachable from the initial marking
  - Transition-system transitions: PN transitions that can fire when being in that marking





Reachability graph

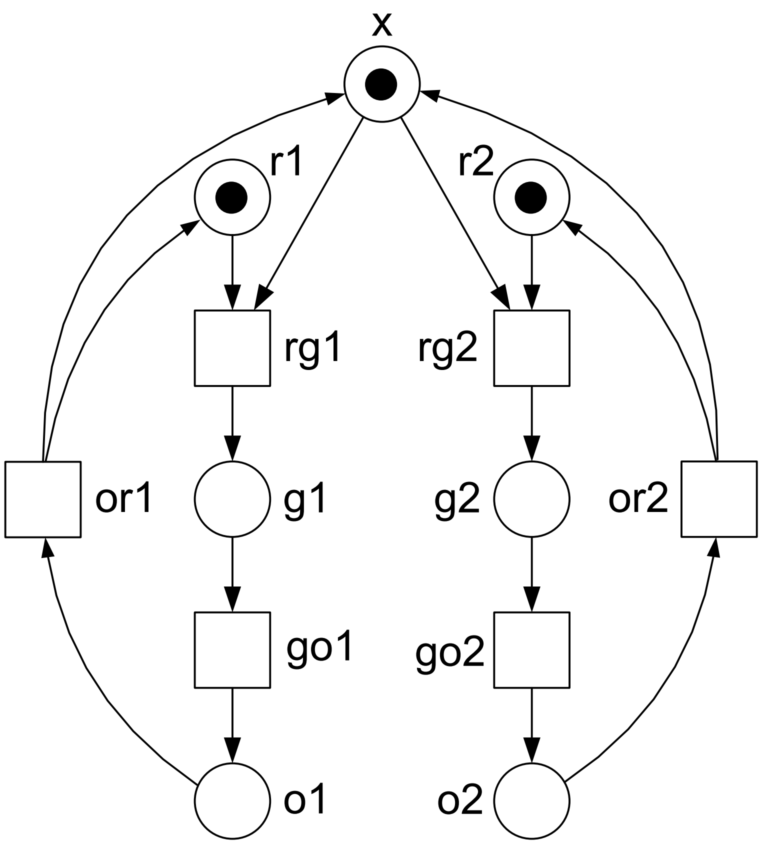


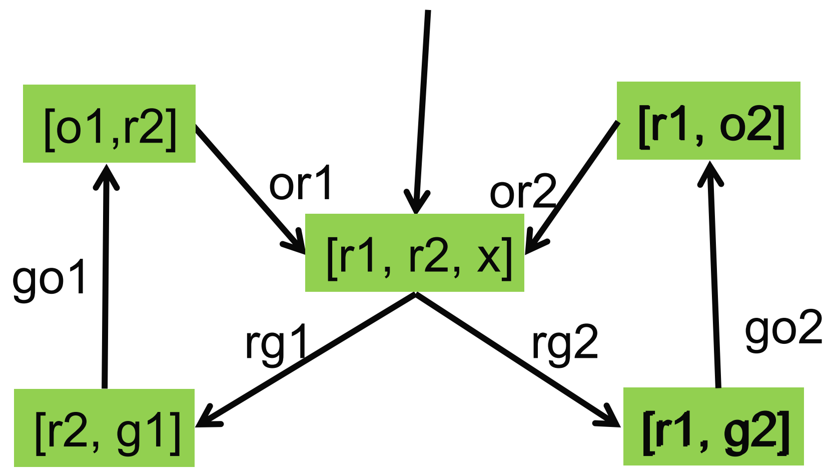
Reachability graph is the *reachable portion* of the transition system (reachable from initial marking)

Reachability Graph Algorithm

1) Label the initial marking m_0 as the *root* and tag it "new".
2) While "new" markings exists, do the following:
   a) Select a new marking m.
   b) If no transitions are enabled at m, tag m "dead-end".
   c) While there exist enabled transitions at m, do the following for each enabled transition t at m:
      i. Obtain the marking m' that results from firing t at m.
      ii. If m' does not appear in the graph add m' and tag it "new".
      iii. Draw an arc with label t from m to m' (if not already present).
3) Output the graph

Example


x
r1
r2
rg1
rg2
or1
g1
g2
or2
go1
go2
o1
o2


[o1,r2]
or1
or2
[r1, o2]
[r1, r2, x]
go1
rg1
rg2
go2
[r2, g1]
[r1, g2]

Example

An important question in Petri net theory is the following:

- Given a Petri net in a marking m_1. Is it possible to execute a sequence of transitions to reach marking m_2. If so, then m_2 is reachable from m_1.

- In more general terms: Given a Petri net in a marking m_1. Is it possible to execute a sequence of transitions to reach some marking m_2 with a specific property p.

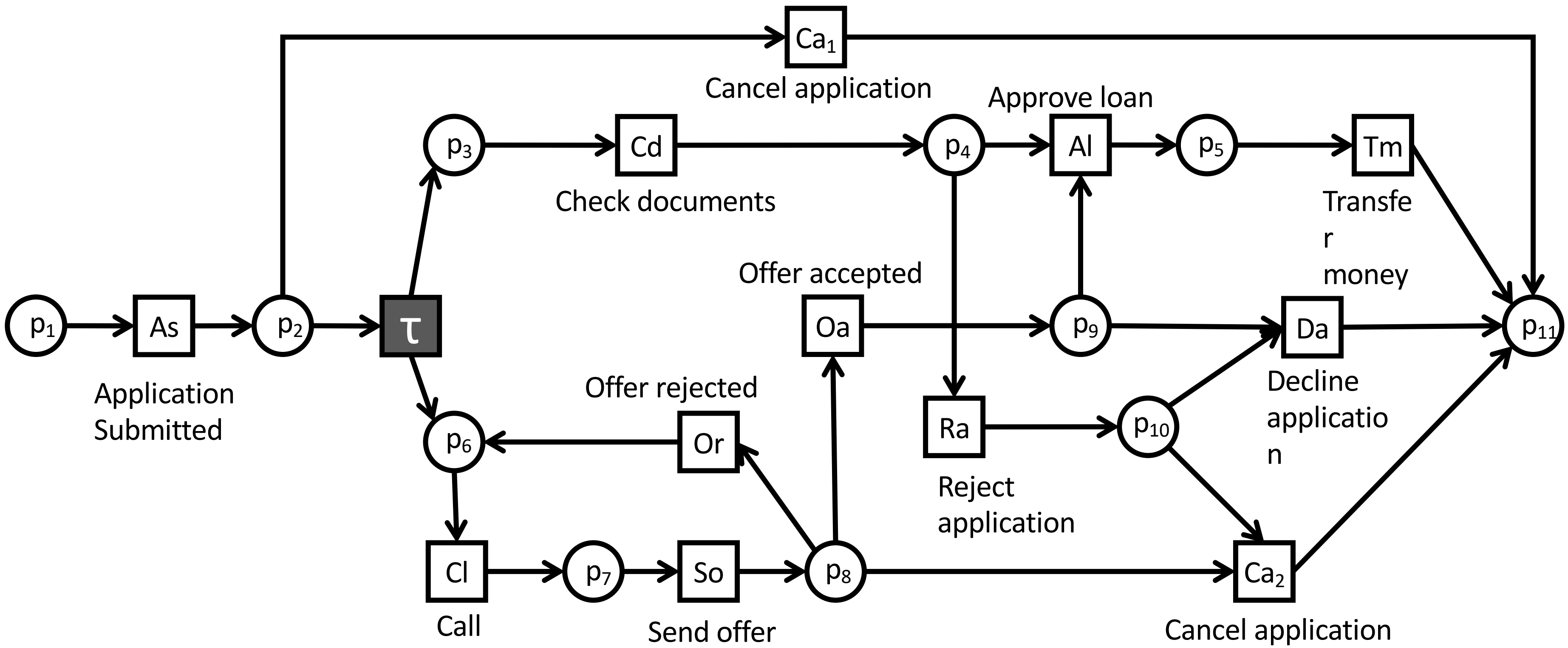
Reachability Questions

- **Boundedness**: Is the number of tokens in each place p less than k in all reachable markings m_2 (for all m_2 and p holds $m_2(p) \leq k$)
- **Dead transition**: Is it possible to reach a marking m_2 in which transition t is enabled (or $m_2 \geq \bullet t$)? If so, t is not dead.
- **Deadlock**: Is it possible to reach a marking in which no transition is enabled (or for all transitions t, $m_2 < \bullet t$)
- **Reversibility:** If the initial marking is m_1, does it hold that for all reachable markings m_2, we can always reach m_1 again?
- **Liveness:** For every reachable marking m_1 and transition t, Is it possible to reach a marking m_2 in which t is enabled ($m_2 \geq \bullet t$)

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Campus d'Excel·lència Internacional

Recall our example

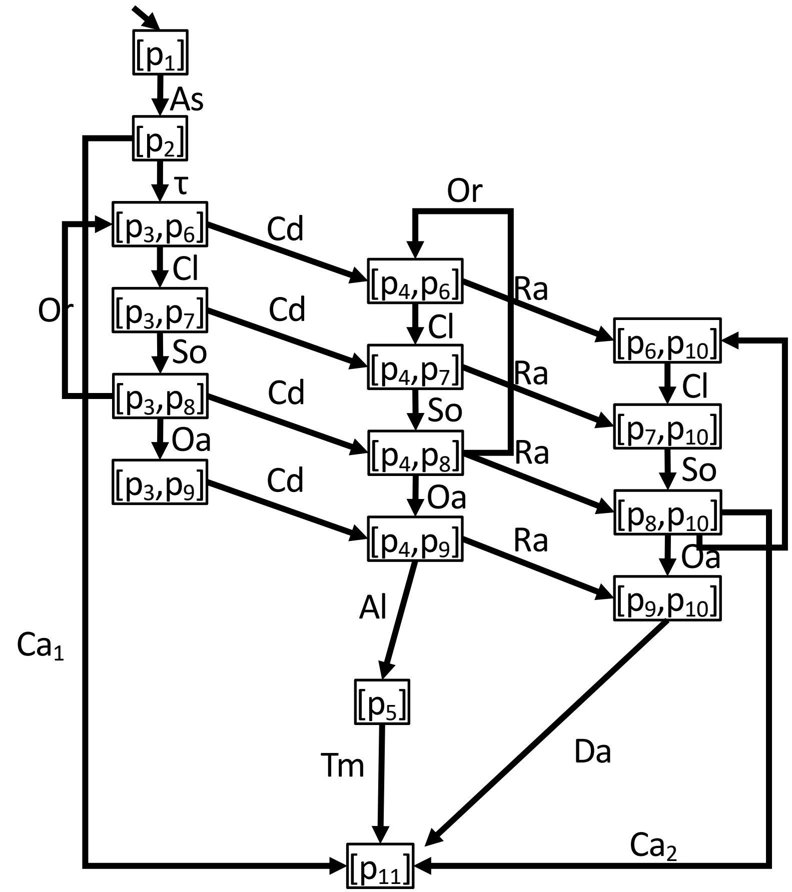


UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

Campus d'Excel·lència Internacional

Reachability Graph

- Deadlocks?
  - YES: [p11]
- Bounded?
  - YES: k=1
- Dead transitions?
  - No
- Reversible?
  - No: [p1] can never be reached again
- Liveness?
  - No: there is a deadlock



Exercise

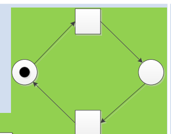
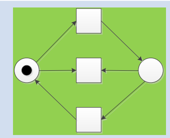
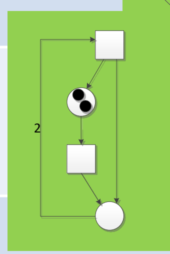
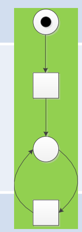
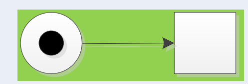
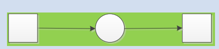
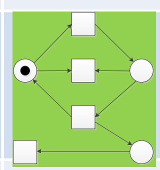
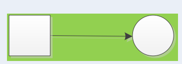
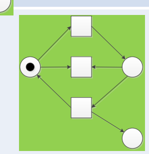
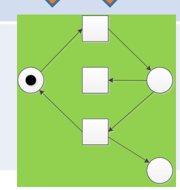
Live (L), Deadlock Free (DF),Bounded (B), Reversible (R)

	Live and Deadlock Free (L,DF)	Live and not Deadlock Free (L,NDF)	Not Live and Deadlock Free (NL,DF)	Not Live and Not Deadlock Free (NL,NDF)
Bounded and Reversible (B,R)				
Bounded and Not Reversible (B,NR)				
Not Bounded and Reversible (NB,R)				
Not Bounded and Not Reversible (NB,NR)				

provide marked Petri net for each cell (if possible)

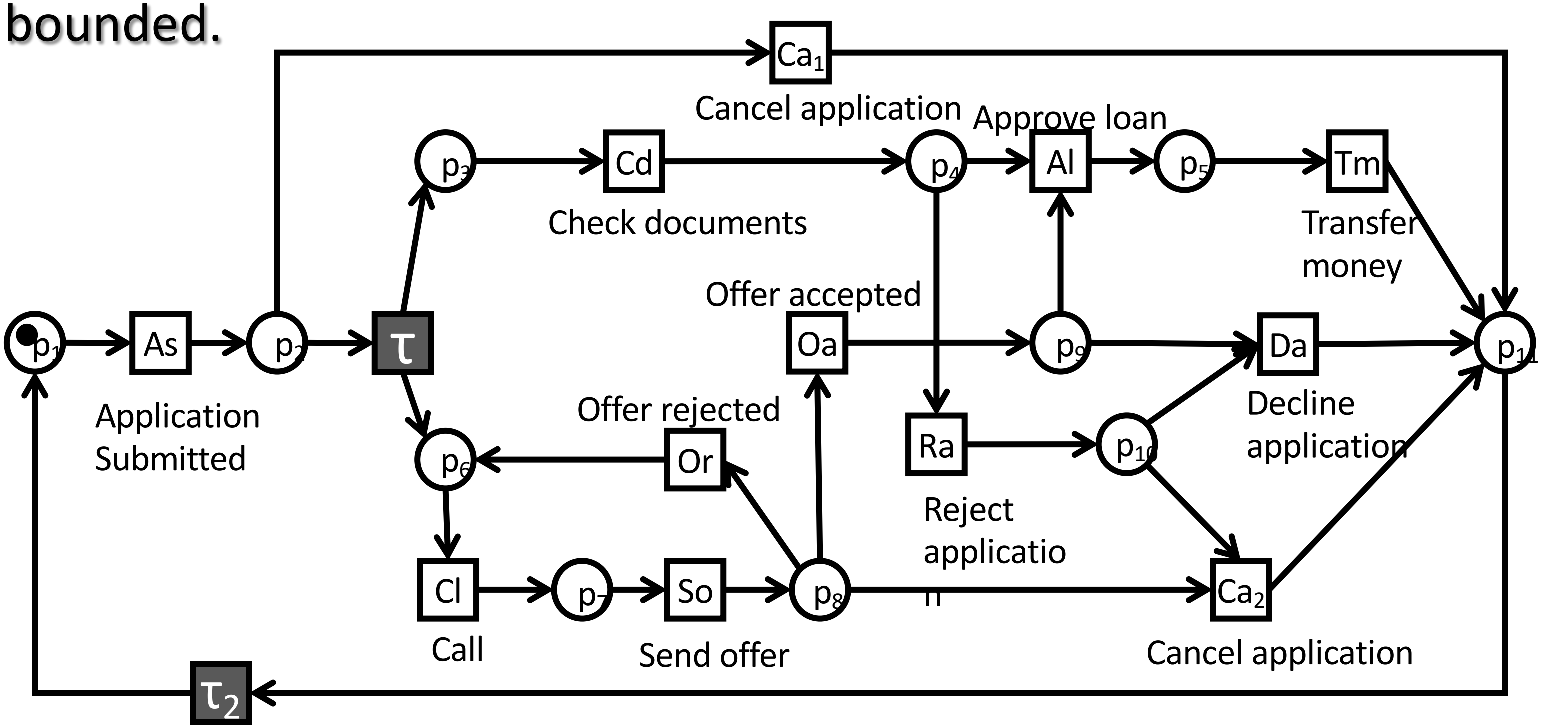
Exercise

Live (L), Deadlock Free (DF),Bounded (B), Reversible (R)

	Live and Deadlock Free (L,DF)	Live and not Deadlock Free (L,NDF)	Not Live and Deadlock Free (NL,DF)	Not Live and Not Deadlock Free (NL,NDF)
Bounded and Reversible (B,R)				
Bounded and Not Reversible (B,NR)				
Not Bounded and Reversible (NB,R)				
Not Bounded and Not Reversible (NB,NR)				

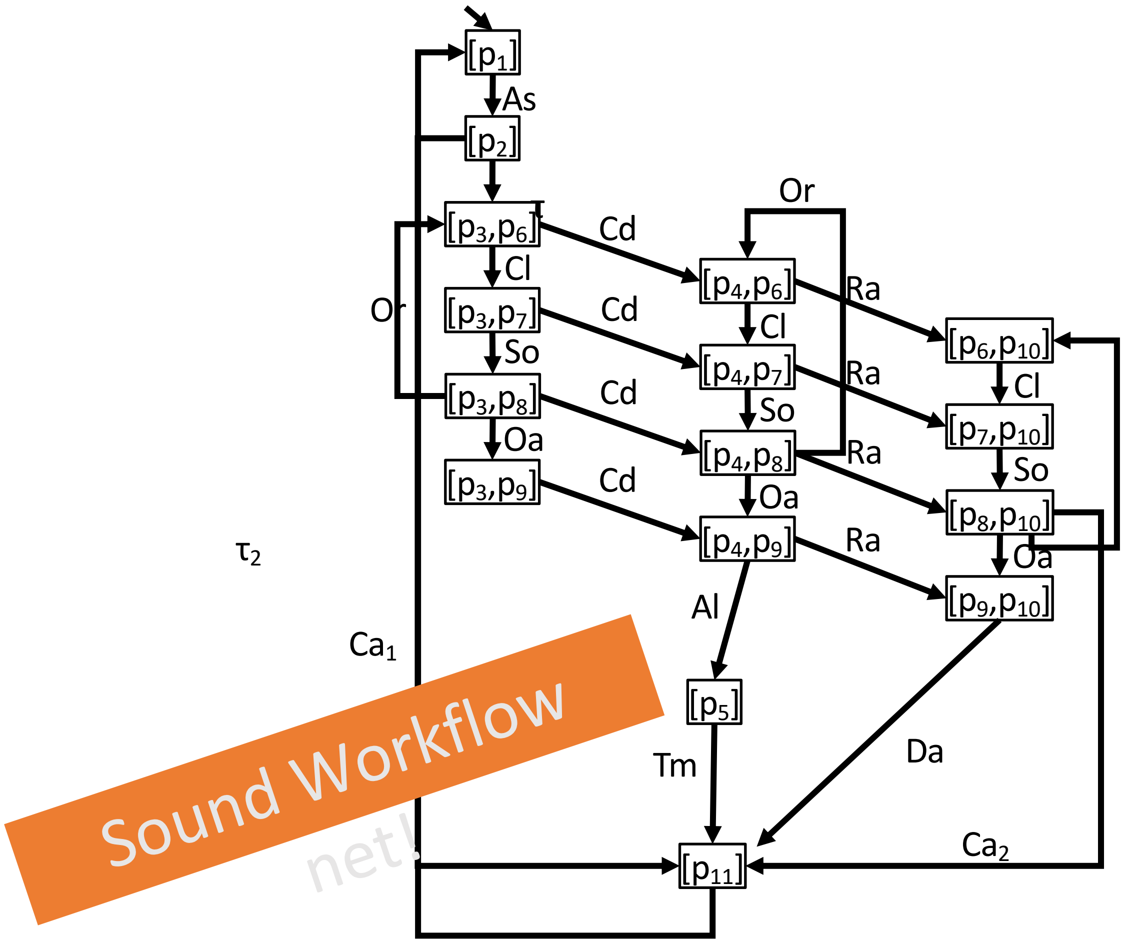
Soundness

A workflow model is sound, if and only if the short-circuited net with marking *[p_1]* is
live and bounded.



Soundness

- Deadlocks?
  - No
- **Bounded?**
  - **Yes**
- Dead transitions?
  - No
- Reversible?
  - Yes
- **Liveness?**
  - **Yes**



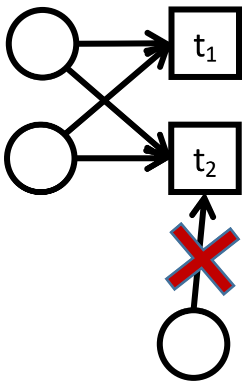
Complexity of Reachability

- Reachability (can a marking be reached?) is a very difficult question.
- The reachability graph can is infinite for unbounded Petri nets.
- For general Petri nets, the reachability question is EXPSPACE- COMPLETE (you need an amount of memory that is worst case exponential in the size of the model.
- Complexity of Soundness is non-primitive recursive…

Petri net Special Cases

- Free Choice Petri Nets
- Block-structured Petri Nets

Free-choice Petri net



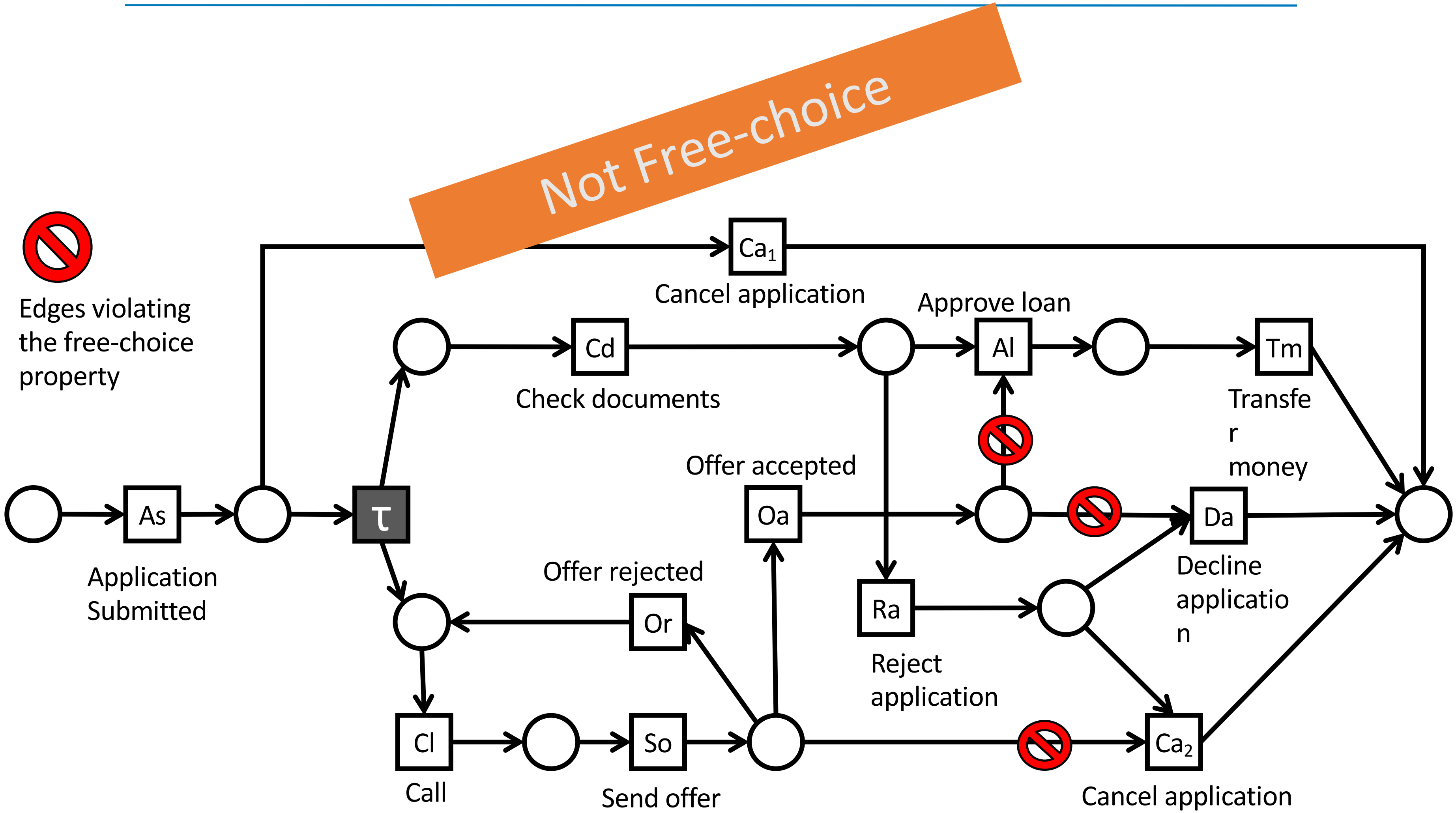
- A Petri net (P,T,F) is free-choice net if and only if $\forall\ t_1,t_2 \in T$ either $\bullet t_1 = \bullet t_2$ or $\bullet t_1 \cap \bullet t_2 = \emptyset$

- Non-free-choice nets should be avoided as much as possible in Business Process Modelling:
  - The behavior depends on the order in which transitions enabled at the same time fire.
  - For most of non-free-choice nets, there is a free-choice net that preserves the same behaviour.

UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

Campus d'Excel·lència Internacional

Free-choice ?



Free-choice Petri Nets Properties

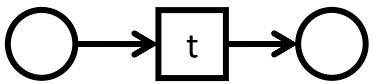
- Free choice nets form the basis of languages like BPMN
- Liveness and boundedness can be decided in polynomial time in the size of the model
- Free choice nets are easy to understand as the behavior is local (all decisions are made locally by transitions sharing input places

Only for Sound Free-choice Workflow models!

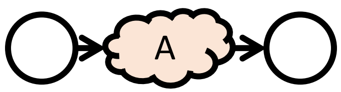
Block-structured Petri Nets

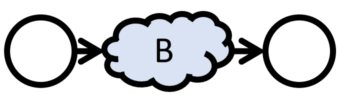
- A special case of Free-choice Petri nets are block-structured Petri nets.
- The basis of a block-structured Petri net is a trivial workflow net with a single transition
- A block-structured Petri net is composed of a hierarchy of block-structured Petri nets by merging input and output places

Block-structured Petri Nets

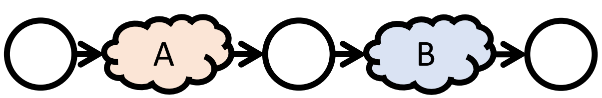
- Basis: 

- Composition rules:

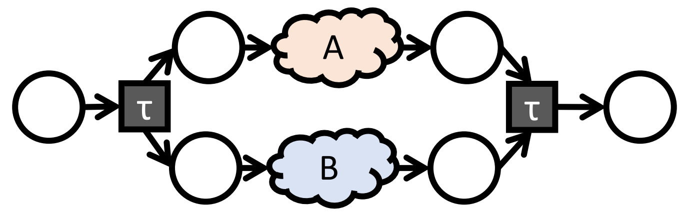
  given: 

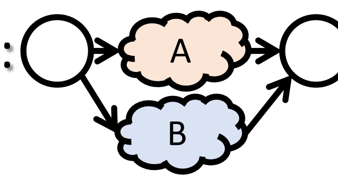
  and: 

- Sequence composition:

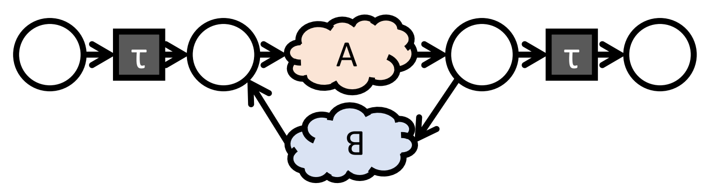


- Parallel composition:



- Choice composition: 

- Loop composition



Take Away

Petri nets is a mathematically well-defined formalism for reasoning about systems. They can be used as internal representation of more user-friendly languages, e.g., BPMN.

Soundness is an important property to guarantee a well-behaved system

Other properties are important

Subclasses of Petri nets