

Business Process Modeling, Management and Mining

Business Process Model and Notation

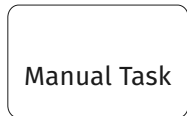
Prof. Cesare Pautasso

<http://www.pautasso.info>

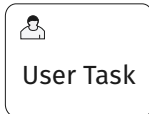
cesare.pautasso@usi.ch

[@pautasso](#)

Human Tasks



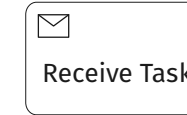
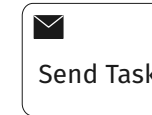
Manual Tasks are carried out by people without using any software system



User Tasks are carried out by people interacting with the workflow engine

Messaging Tasks

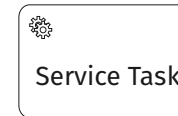
Represent message exchanges between participants



Send Tasks will complete once the message is sent

Receive Tasks will wait for a message and complete once it arrives

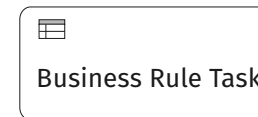
Automated Tasks



Web Service invocations,
External Software Applications

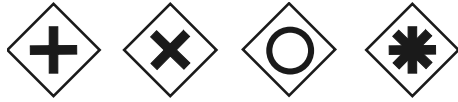


Local computations (scripts
embedded within the Workflow
engine)



Evaluation of business rules
(e.g., to make decisions)

Control Flow Gateways



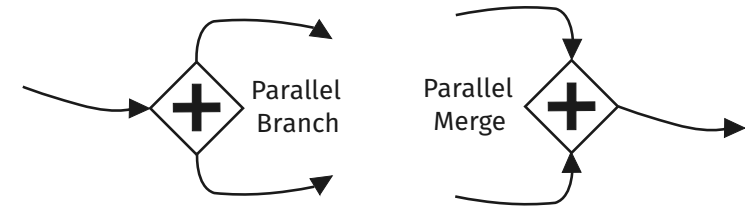
Gateways are a syntactic construct to represent branches and merges in the control flow graph

Gateways are not labeled with any "business" information (use events and activities instead)

Gateways represent decision points and can be annotated with conditions

AND Gateway

Parallel gateways represent parallel fork/joins in the control flow

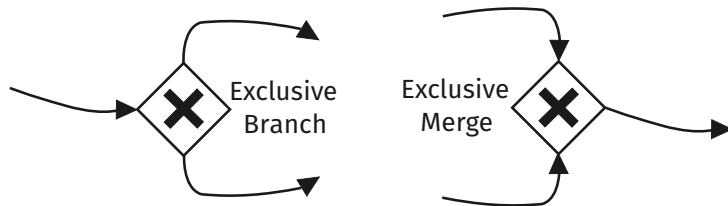


Branch: a token is sent on every outgoing control flow edge

Merge: wait for a token to arrive from each incoming edge

XOR Gateway

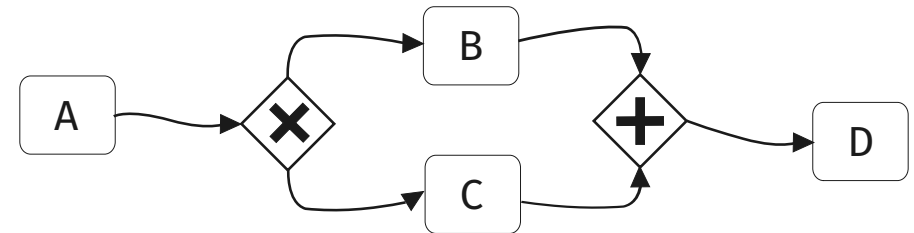
Exclusive gateways represent sequential branches/merges in the control flow



Branch: every incoming token corresponds to at most one outgoing token (statically checked)

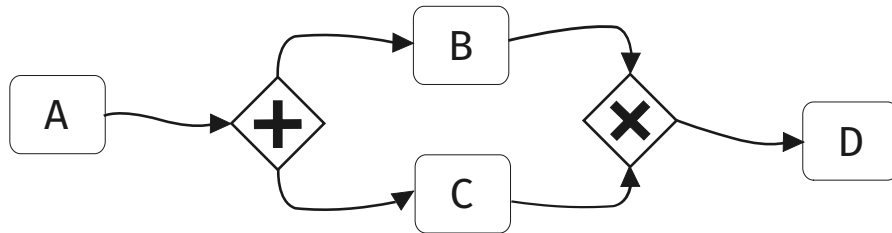
Merge: no synchronization is performed with multiple incoming tokens

Deadlock



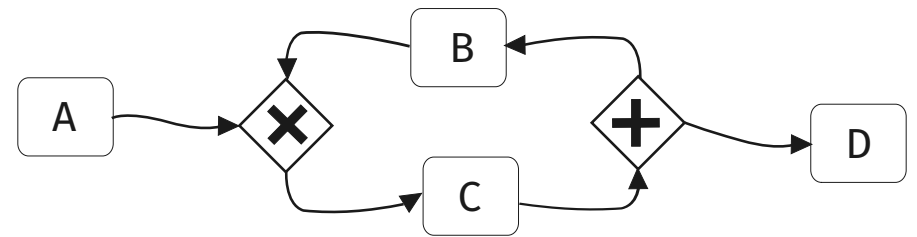
The merge AND gateway will wait forever for 2 tokens, while the branch XOR gateway will never produce more than one

Lack of Synchronization



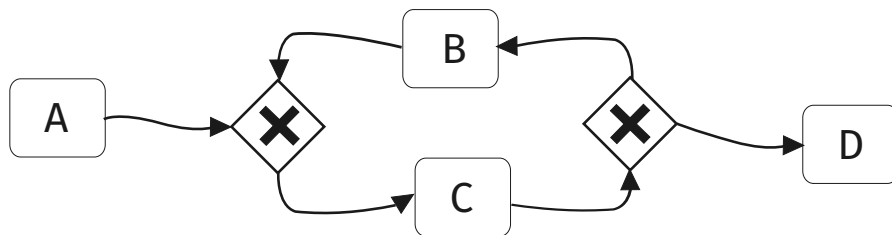
The merge XOR gateway will let through both tokens produced by the AND parallel branch

Livelock



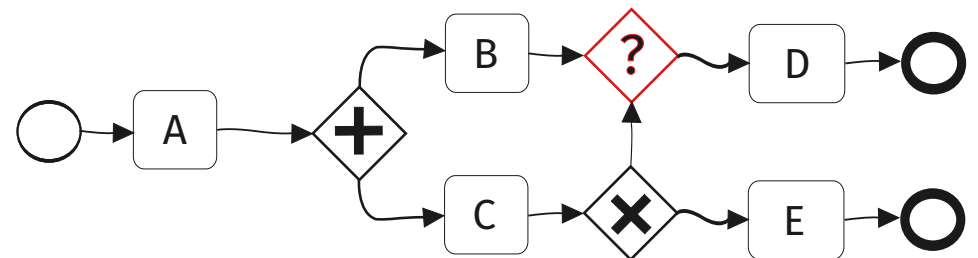
The loop will never exit, and whatever follows (D) the cycle in the graph will also be repeated with every iteration

Loop



The merge XOR gateway will let through any incoming token, either from upstream A or from the outgoing XOR branch. The loop will exit once control flow reaches D.

Which gateway do we need?

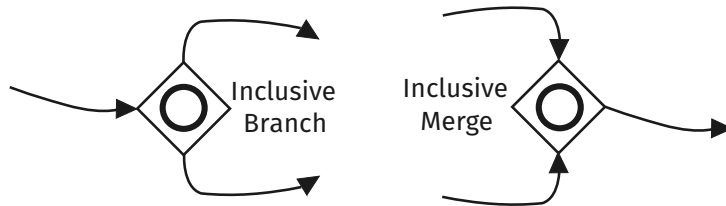


AND merge would result in a potential deadlock

XOR merge would result in a potential lack of synchronization

OR Gateway

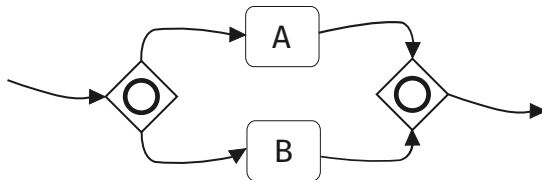
Inclusive gateways represent overlapping sequential/parallel branches



Branch: a token is sent on every outgoing control flow edge with a satisfied condition

Merge: wait for a token to arrive from each edge that was triggered in the inclusive branch

OR Gateway Equivalent Semantics



Can you build an equivalent control flow graph using only XOR and AND gateways?