The Application of Petri Nets to Workflow Management by W.M.P. van der Aalst

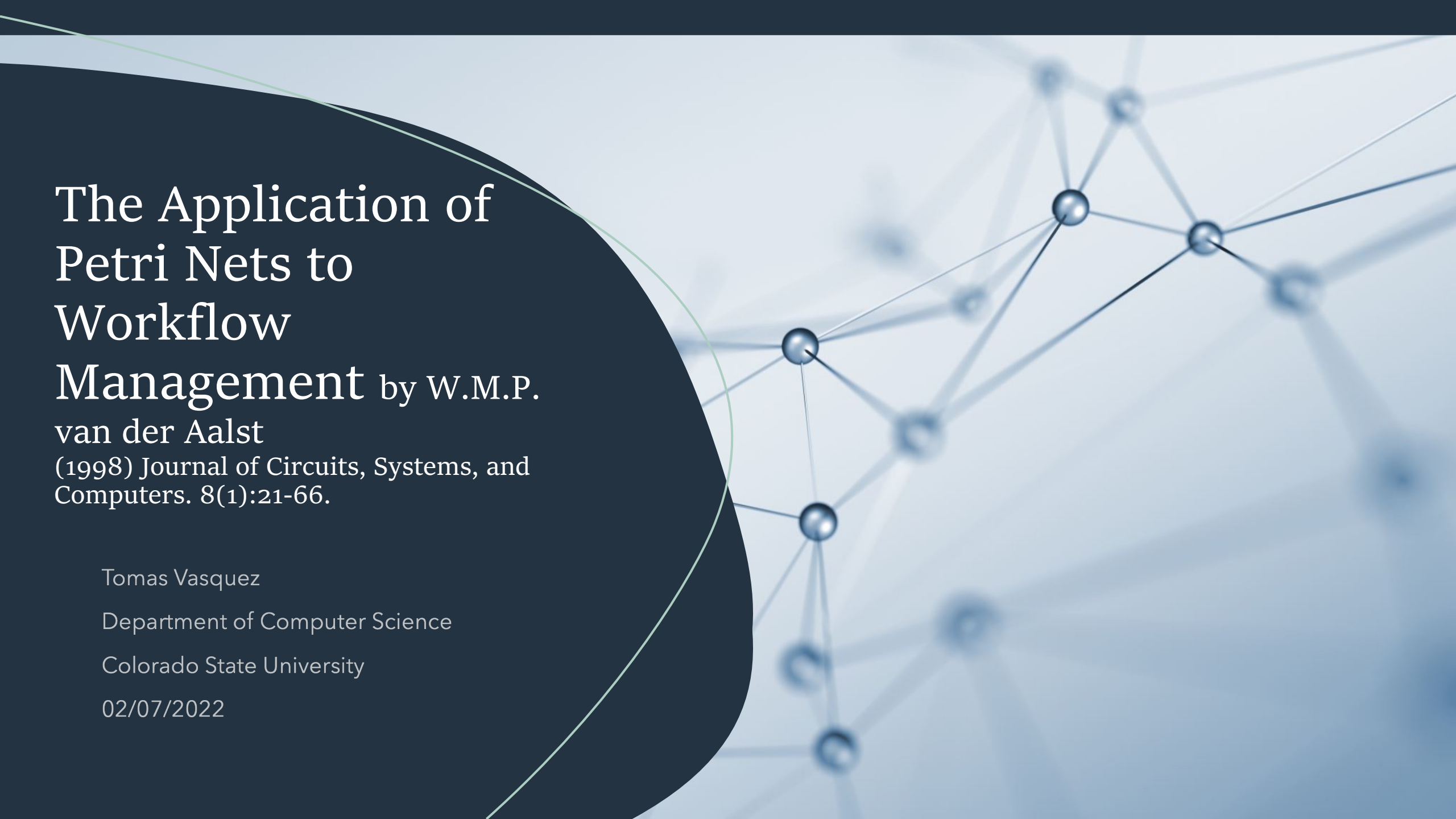
(1998) Journal of Circuits, Systems, and Computers. 8(1):21-66.

Tomas Vasquez

Department of Computer Science

Colorado State University

02/07/2022



Introduction

Problem Statement:

- This paper illustrates the application of Petri Nets to workflow management.
- How to Analyze the correctness of workflows using Petri Nets and their properties.
- Discussion of Tools for workflow management.

All materials, ideas, and concepts were taken from [1] unless specified otherwise.

Agenda

- Workflow and Workflow management definitions
- Example of a Petri Net
- High Level Petri Nets
- Workflow Routing operators on Petri Nets
- Correctness and Soundness Property
- Limitations of Petri Nets and efficient subclasses of nets
- Resiliency

All materials, ideas, and concepts were taken from [1] unless specified otherwise.

What is a workflow?

- A workflow is a 3-dimensional construct relating cases, resources, and processes.
- Three characteristics of a workflow:
  - Case-driven, essential, and it can be defined in an explicit manner.

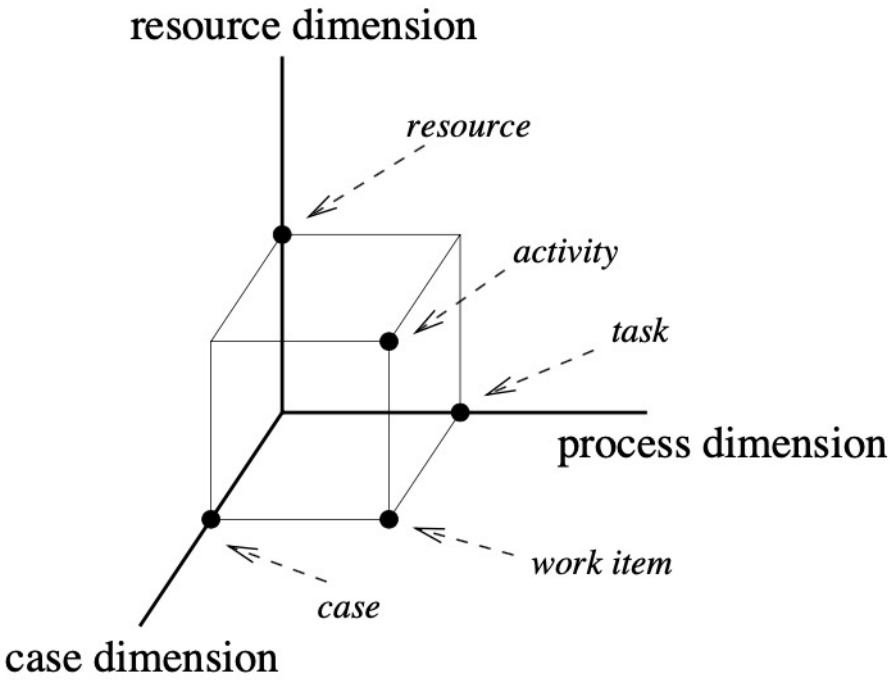


Figure 2: A three dimensional view of a workflow.

[1]

Sample Workflow:

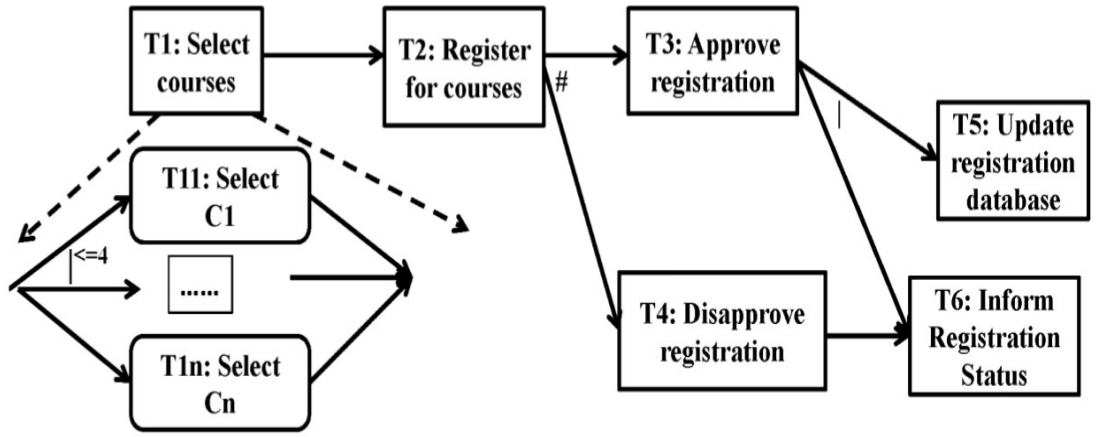


Fig. 1. Sample workflow for course registration.

$(u_1,\text{GradStudent}) \in UA$, $(u_2,\text{GradDirector}) \in UA$,
$(u_3,\text{RegistrationOfficer}) \in UA$

$(T_1,\text{GradStudent}) \in PA$, $(T_2,\text{GradStudent}) \in PA$,
$(T_3,\text{GradDirector}) \in PA$, $(T_4,\text{GradDirector}) \in PA$,
$(T_5,\text{RegistrationOfficer}) \in PA$, $(T_6,\text{RegistrationOfficer}) \in PA$

$=(T_1, T_2), \neq(T_2, T_3), \neq(T_2, T_4), \text{Cardin}(u_1, 4)$

[2]

All materials, ideas, and concepts were taken from [1] unless specified otherwise.

Workflow Management System:

- A workflow management system (WFMS) is a generic software tool that allows for the definition, execution, registration, and control of workflows[1].

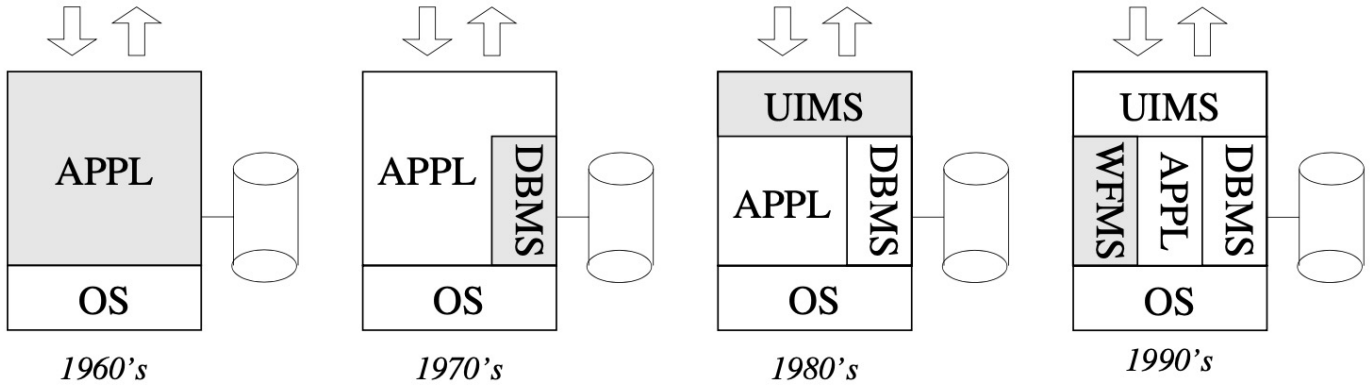


Figure 1: Workflow management systems in a historical perspective.

[1]

Example of a Petri Net:

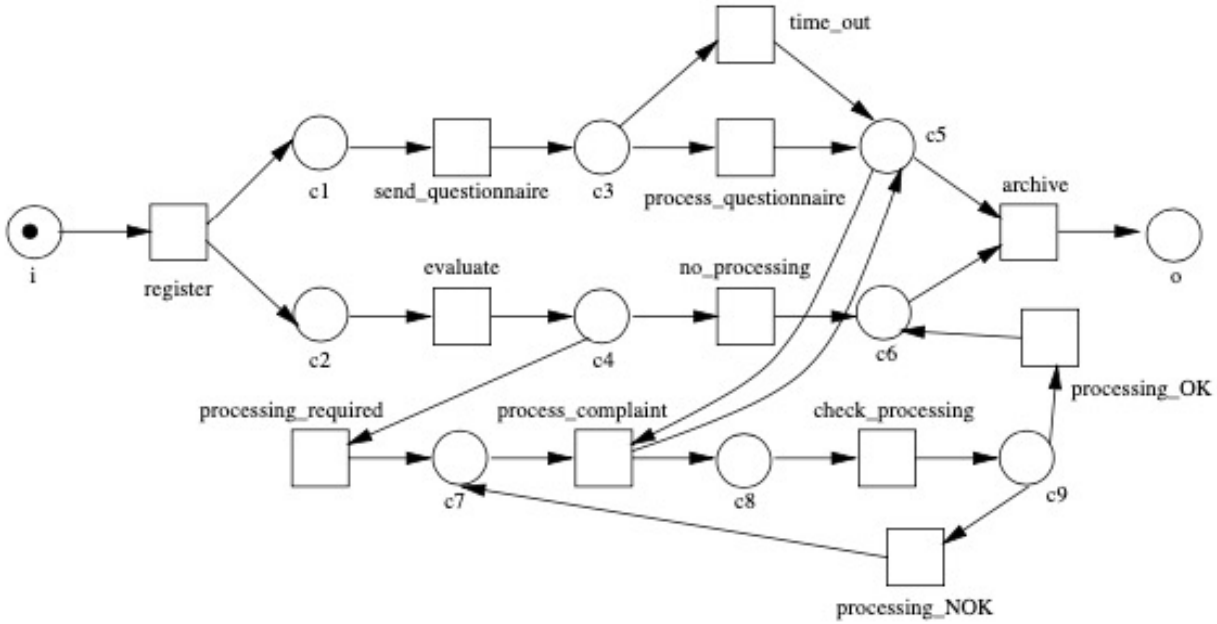


Figure 6: A Petri net for the processing of complaints.

[1]

Example of a More Complicated Petri Net

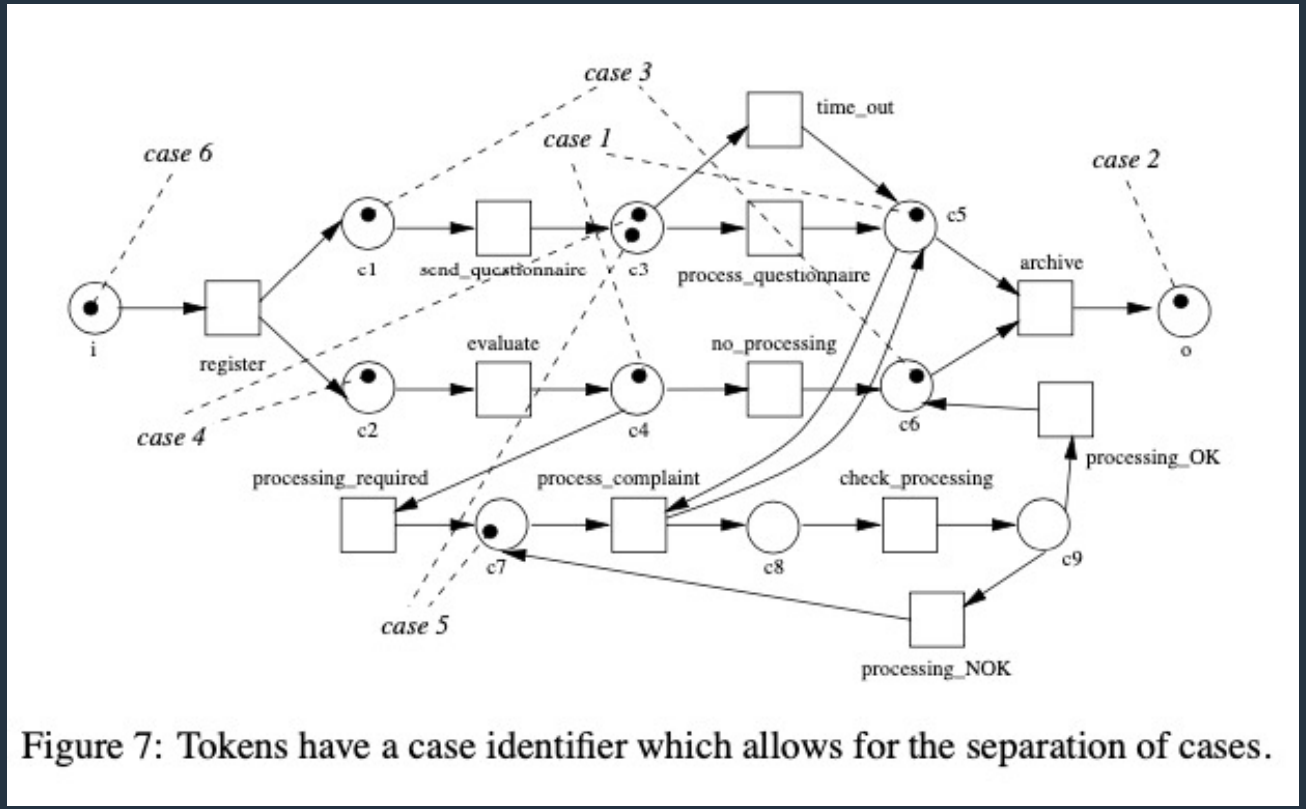


Figure 7: Tokens have a case identifier which allows for the separation of cases.

[1]

High Level Petri Nets

- Color
  - Tokens can represent different objects with attributes.
- Time
  - Describes temporal behavior of the system which may be associated with places, transitions, and/or tokens.
- Hierarchy
  - Allows for subnets that can be combined to form large and complex systems.

Enabling and Firing of Transitions

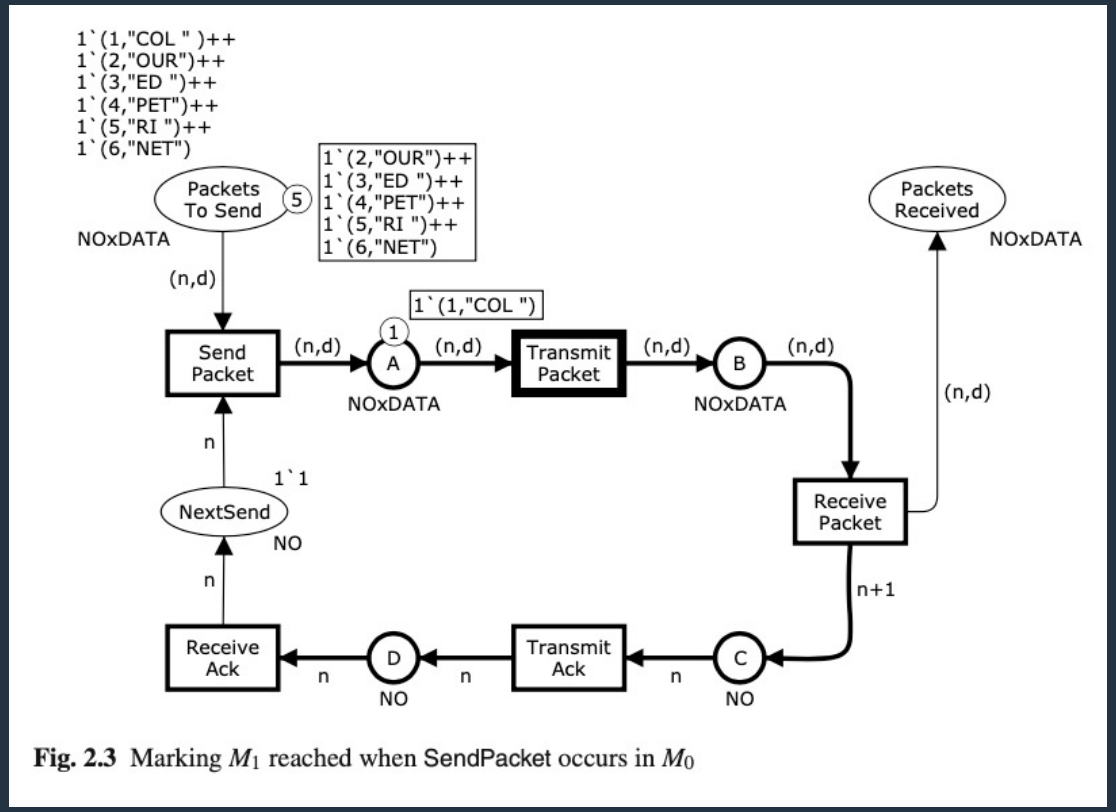


Fig. 2.3 Marking M_1 reached when SendPacket occurs in M_0

[4]

The Four Workflow Routing Types

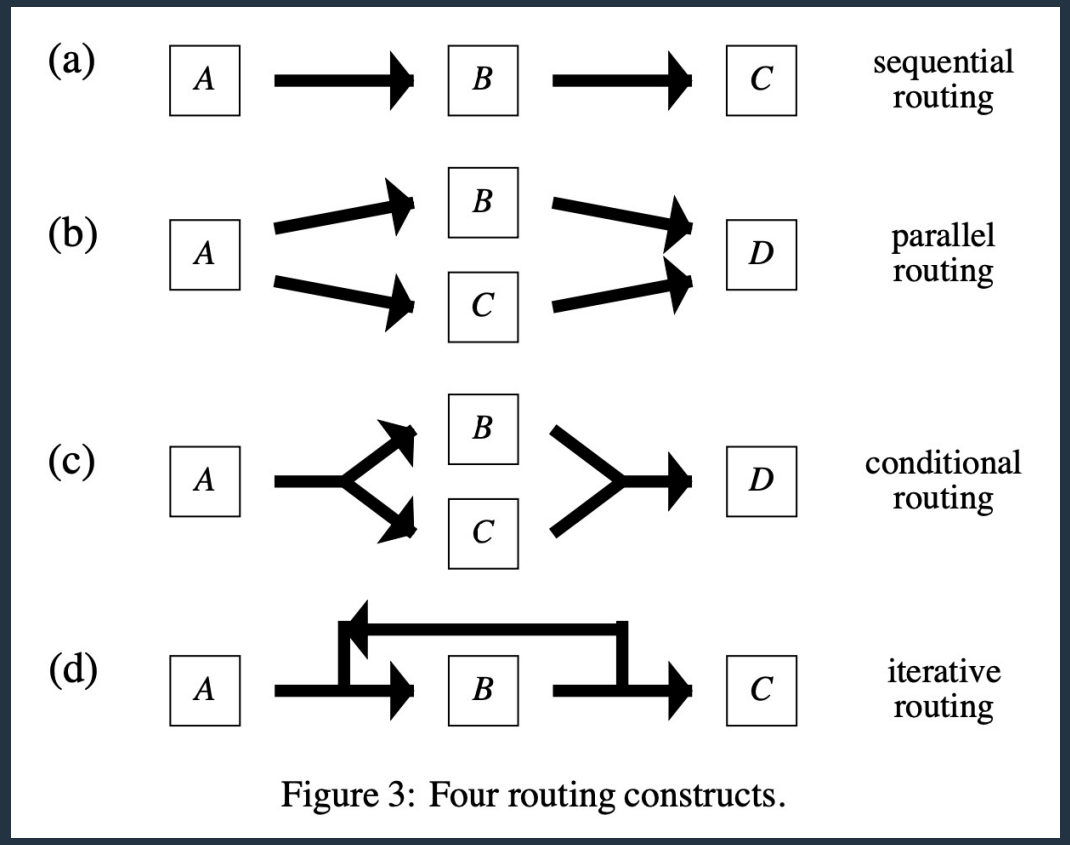


Figure 3: Four routing constructs.

[1]

- Work- flow Management Coalition (WfMC) defines four routing types:
- Sequential
- Parallel
  - Non-Deterministic
- Conditional equivalent to Exclusive OR
  - Non-Deterministic or Deterministic
- Iteration
- Mapped to Control Tasks

All materials, ideas, and concepts were taken from [1] unless specified otherwise.

Sequential Routing

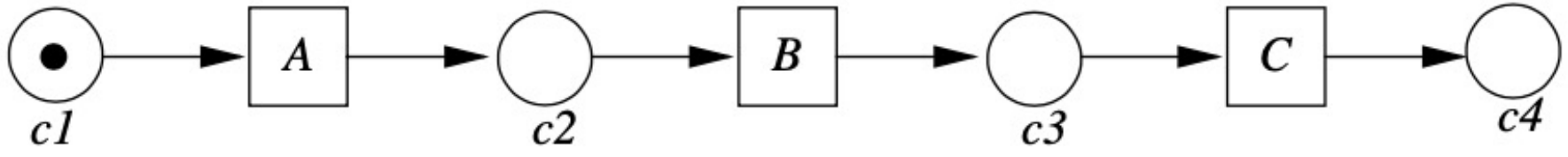
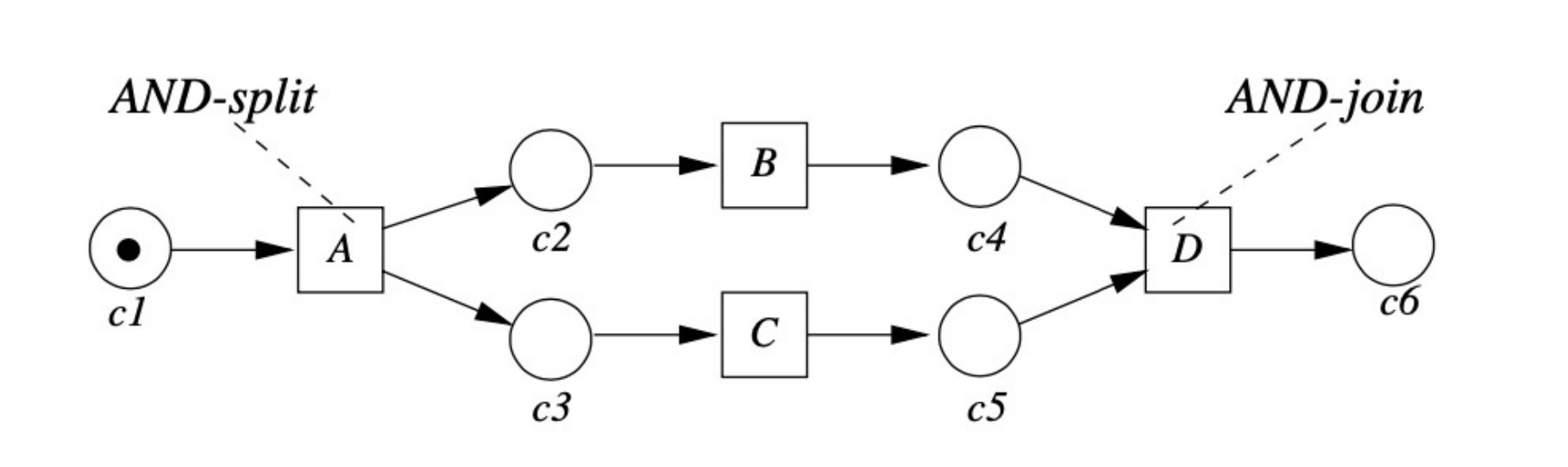


Figure 8: Sequential routing.

[1]

Parallel Routing



[1]

Conditional Routing/Implicit “OR”/Non-deterministic “OR”

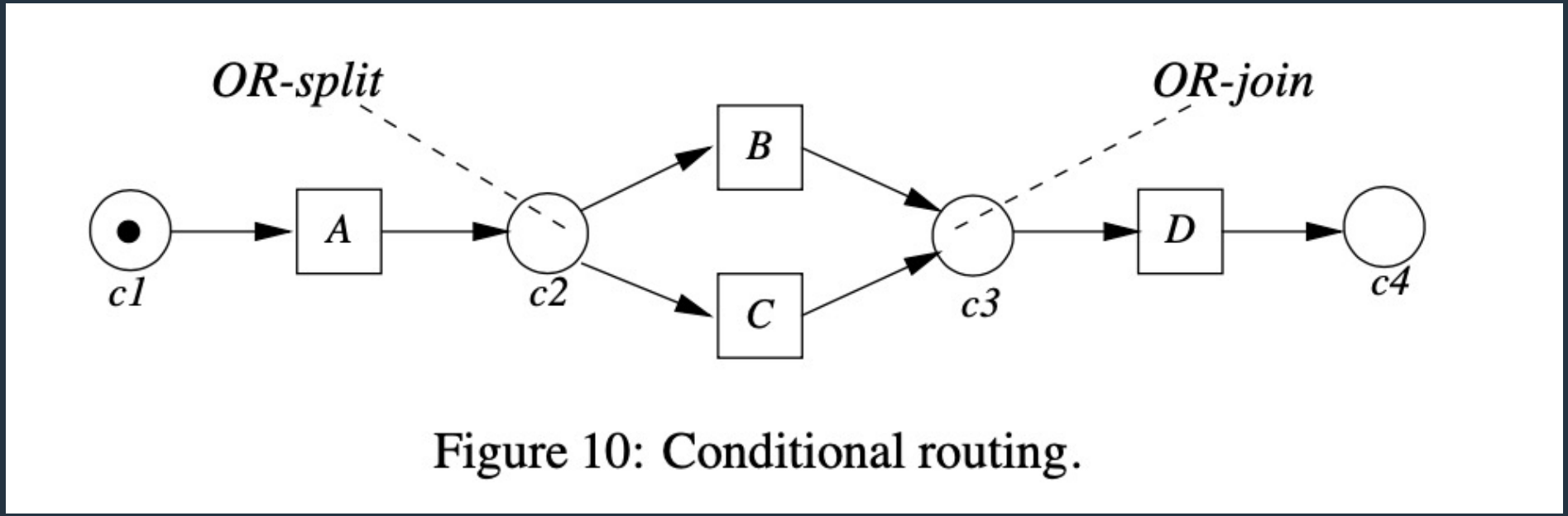


Figure 10: Conditional routing.

[1]

Conditional Routing/Explicit "OR"/Deterministic "OR"

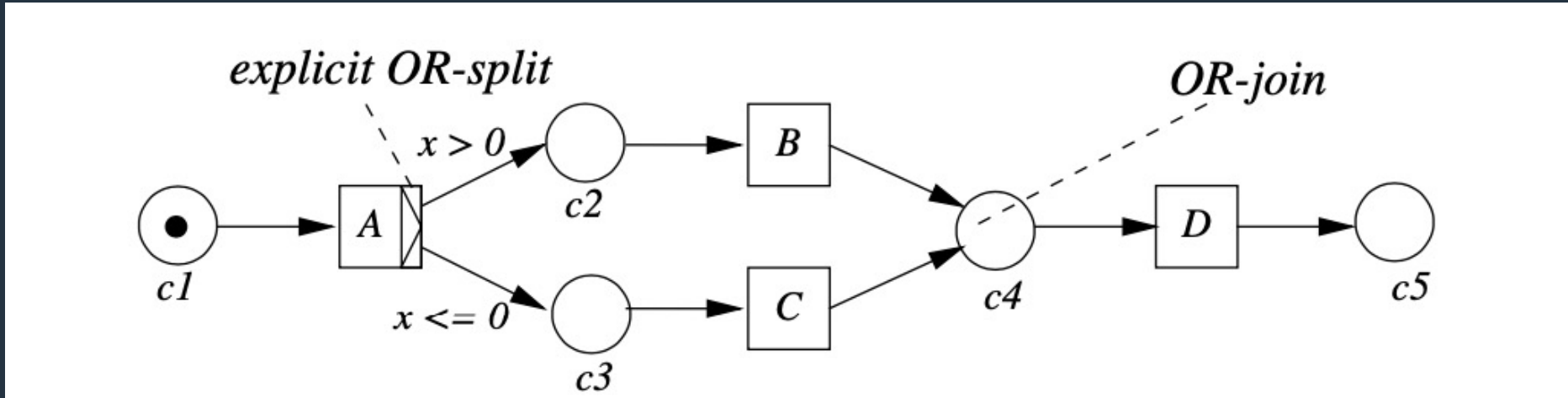


Figure 11: Explicit choice between B and C based on workflow attribute x.

[1]

Looping

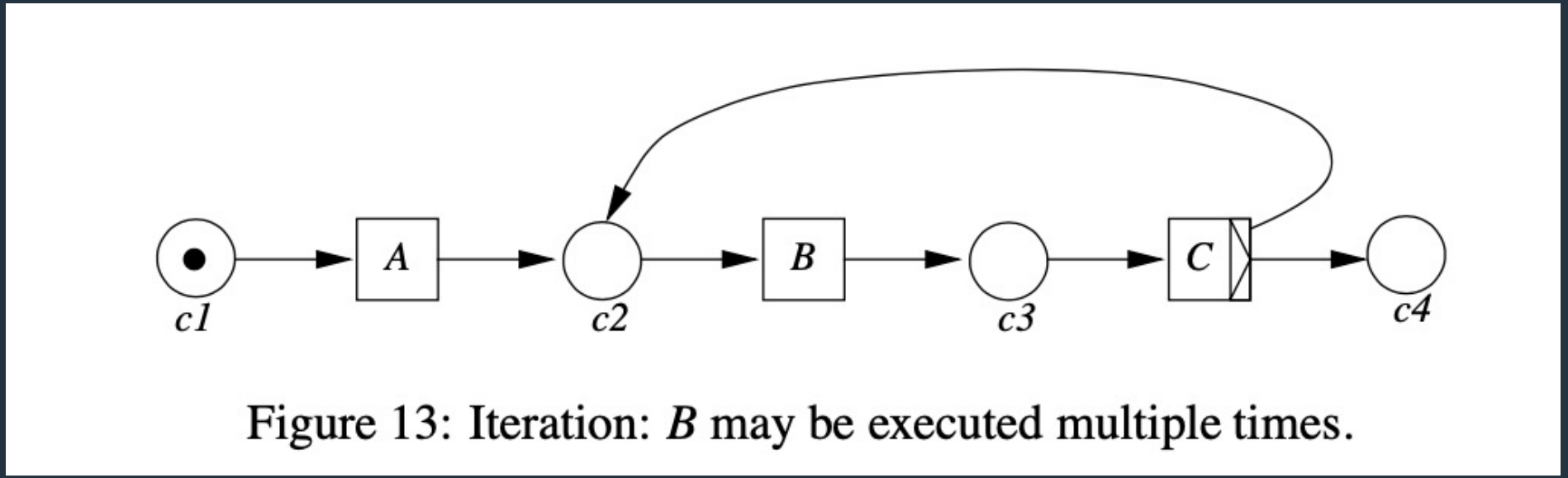


Figure 13: Iteration: *B* may be executed multiple times.

[1]

What attributes can we evaluate in a workflow?

- Correctness
- Effectiveness
- Efficiency
- What about Resilience?

What does it mean for a workflow to be correct?

- A workflow should have a source place and a sink place.
- Each task and pre/post condition must be on a path from the source place to the sink place.
- For any case, the procedure will terminate eventually
- There should be no dead tasks.

Workflow Net

- Models one case in isolation
- One unique input/output place
- Strong connectivity of the Extended Net

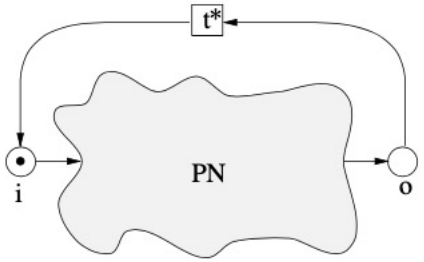


Figure 23: $\overline{PN} = (P, T \cup \{t^*\}, F \cup \{\langle o, t^* \rangle, \langle t^*, i \rangle\})$.

[1]

Soundness property:

- For every state reachable from the input node, there is a path to the output node.
- Once the procedure terminates all places are empty except for the output place.
- No dead transitions

Necessary and Sufficient Conditions for Soundness of WF-net:

- Theorem 1: A WF-net is sound $\Leftrightarrow$ (PN', i) is live and bounded.

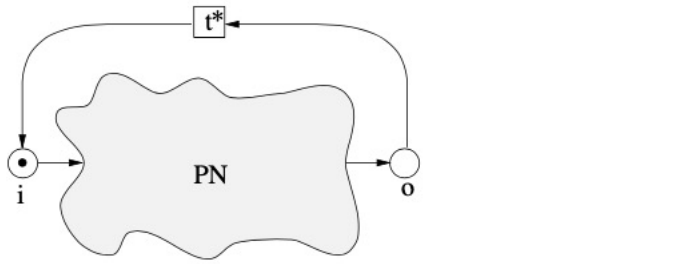


Figure 23: $\overline{PN} = (P, T \cup \{t^*\}, F \cup \{\langle o, t^* \rangle, \langle t^*, i \rangle\})$.

[1]

All materials, ideas, and concepts were taken from [1] unless specified otherwise.

How much better are workflow nets for analysis?

- It depends on the Net.
- Theorem 1 still does not address the following:
  - For a complex WF-net it may be intractable to decide soundness.
  - For arbitrary WF-nets soundness is decidable but also expensive in terms of time and space complexity.
  - Deciding Liveness and Boundness is EXPSAPCE-hard

Free Choice Workflow Nets:

- Main idea: You can not mix choice and synchronization.
- A petri net $PN = (P, T, \mathrm{F})$ is a free choice net $\Leftrightarrow \forall\, t_i t_j \in T,\ {}^{*}t_i \cap {}^{*}t_j \neq \emptyset \Rightarrow t_i = t_j$.
- *May be checked for soundness in polynomial time.*
- Supports parallelism, sequential routing, conditional routing and iteration.

Example of Violating the Free Choice Property:

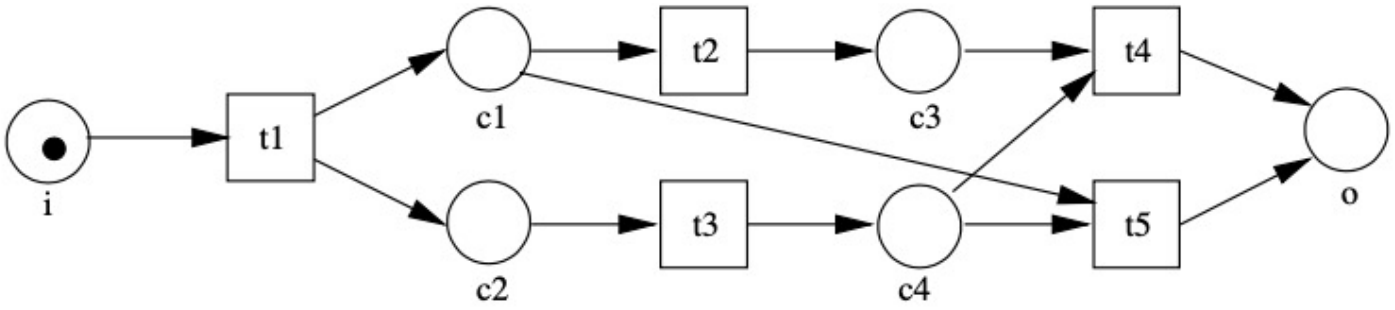


Figure 24: A non-free-choice WF-net containing a mixture of parallelism and choice.

[1]

Well-structured WF-nets

- Main Idea: Balance AND splits/joins and OR split/joins.
- A workflow net PN is well handled ⟺ the extended petri net PN' is well handled.
- *A well-structured WF-net can be checked for soundness in polynomial time.*

Example of a Workflow that is not Well Structured:

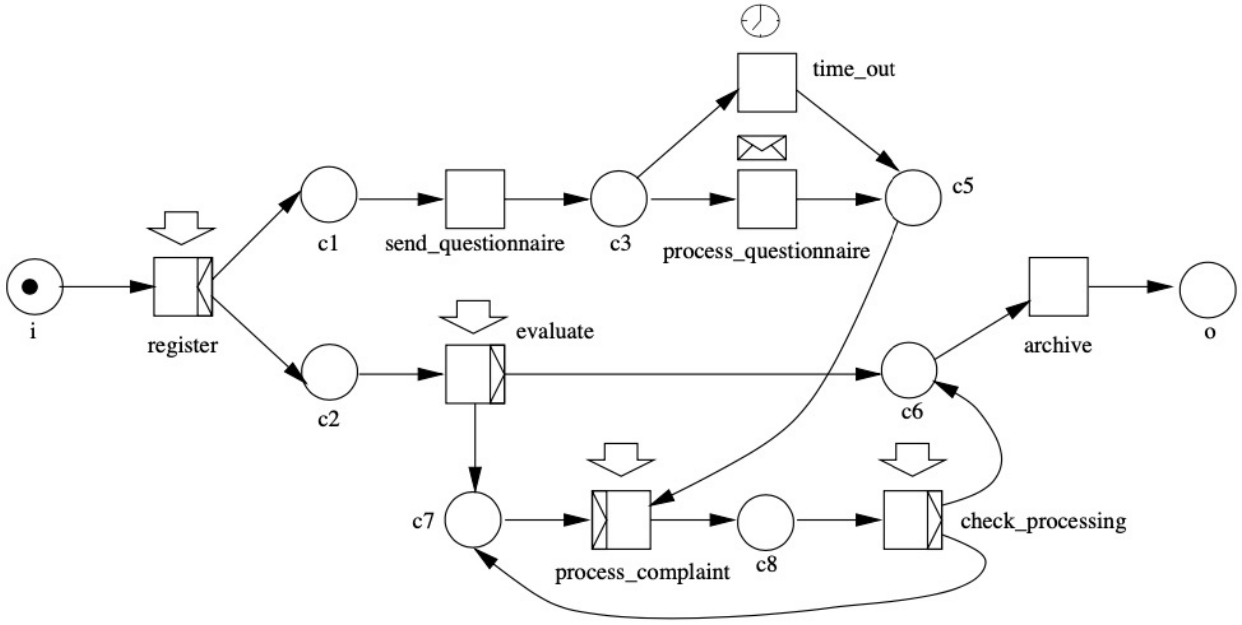


Figure 26: A workflow process definition which is not well-structured.

[1]

Example of a Workflow that is Well Structured but not Free Choice:

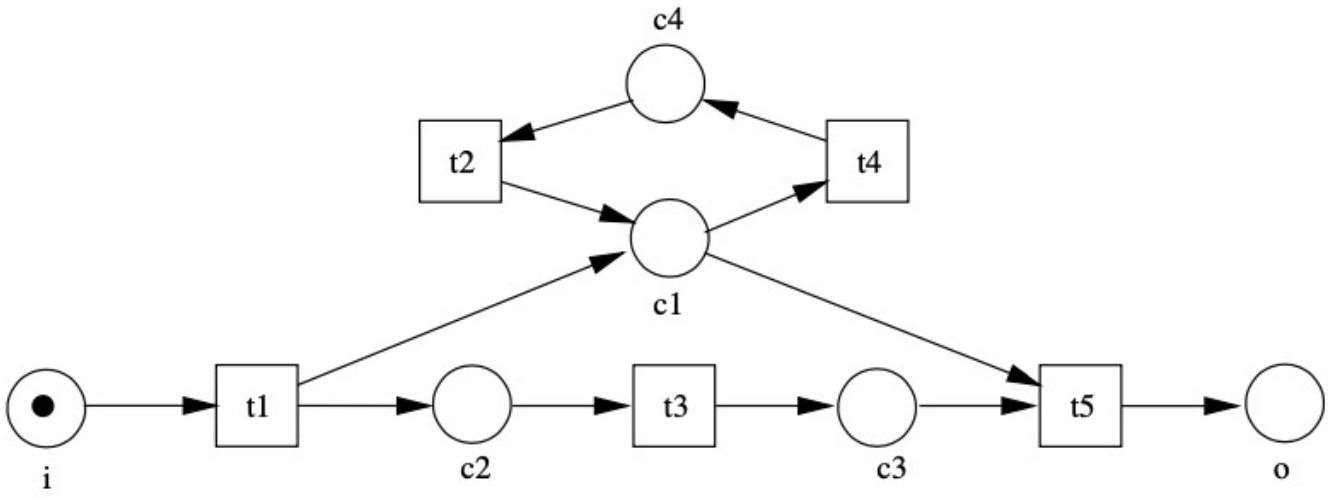


Figure 27: A well-structured WF-net.

[1]

Our Work: Analyzing Resilience

- Include all routing types plus additional types

  $\left|\right|^{k}(W_1, \dots, W_n)\left|\right|$ $|^{\leq k}(W_1, \dots, W_n)|^{\leq k}$ $\left|\right|^{\geq k}(W_1, \dots, W_n)\left|\right|$

  $if\ \{C\}\ then\ W_1\ else\ W_2$ |
- How do the different operators affect the ability of the workflow to terminate given that some task fails?
- How do we quantify it?

All materials, ideas, and concepts were taken from [1] unless specified otherwise.

References

[1] Aalst, Wil. (1998). The Application of Petri Nets to Workflow Management. Journal of Circuits, Systems, and Computers. 8. 21-66. 10.1142/S0218126698000043.
[2] P. Yang, X. Xie, I. Ray and S. Lu, "Satisfiability Analysis of Workflows with Control-Flow Patterns and Authorization Constraints," in IEEE Transactions on Services Computing, vol. 7, no. 2, pp. 237-251, April-June 2014, doi: 10.1109/TSC.2013.31.
[3] Dr. Van der Aalst personal website: http://www.padsweb.rwth-aachen.de/wvdaalst/index.html
[4] K. Jensen and L.M. Kristensen.
Coloured Petri Nets: Modelling and Validation of Concurrent Systems. Springer, 2009.

All materials, ideas, and concepts were taken from [1] unless specified otherwise.

Thank You!

All materials, ideas, and concepts were taken from [1] unless specified otherwise.