

# Accepted Manuscript

Dynamic Skipping and Blocking, Dead Path Elimination for Cyclic Workflows, and a Local Semantics for Inclusive Gateways

Dirk Fahland, Hagen Völzer

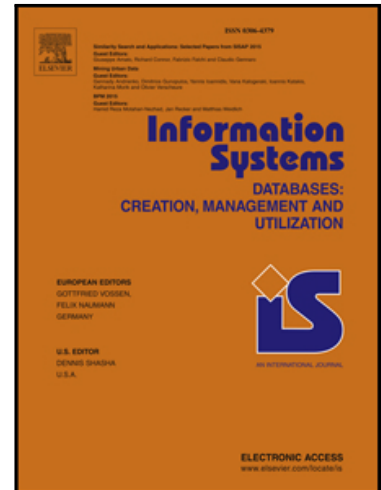
PII: S0306-4379(16)30634-2  
DOI: [10.1016/j.is.2018.04.004](https://doi.org/10.1016/j.is.2018.04.004)  
Reference: IS 1305

To appear in: *Information Systems*

Received date: 21 December 2016  
Revised date: 29 August 2017  
Accepted date: 22 April 2018

Please cite this article as: Dirk Fahland, Hagen Völzer, Dynamic Skipping and Blocking, Dead Path Elimination for Cyclic Workflows, and a Local Semantics for Inclusive Gateways, *Information Systems* (2018), doi: [10.1016/j.is.2018.04.004](https://doi.org/10.1016/j.is.2018.04.004)

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.



## Highlights

- We define dynamic skipping and blocking for BPMN-like languages, each with a dedicated local semantics, such that they can be used independently from each other or freely combined. We define the semantics for general control-flow graphs, including cyclic graphs, and compare the semantics of static and dynamic skipping and blocking.
- Our proposal for dynamic blocking includes a generalization of the Dead-Path-Elimination (DPE) concept to general control flow, which so far was limited to acyclic control flow.
- We point out that dynamic blocking is closely related with the semantics of inclusive gateways (aka synchronizing merge pattern, OR-join semantics). Our generalization of DPE to cyclic flow graphs gives rise to a purely local semantics for inclusive join behavior. As a result, our semantics does not entail semantic anomalies such as vicious cycles. In comparison with existing semantics, it can be enacted faster, i.e., in constant time, it is compositional for more models and therefore easier to understand and use, and it permits more refactoring operations for process models.
- We show that dynamic skipping and blocking can be realized on existing process execution engines. We provide a local syntactic transformation of workflow graphs with skipping and blocking to classical workflow graphs with process variables but without inclusive gateways. The translation permits to enact dynamic skipping and blocking and their application to Dead-Path-Elimination and local semantics for inclusive joins without modifying existing process execution engines.

Download English Version:

<https://daneshyari.com/en/article/9952084>

Download Persian Version:

<https://daneshyari.com/article/9952084>

[Daneshyari.com](https://daneshyari.com)