Dr. Dirk **Fahland**

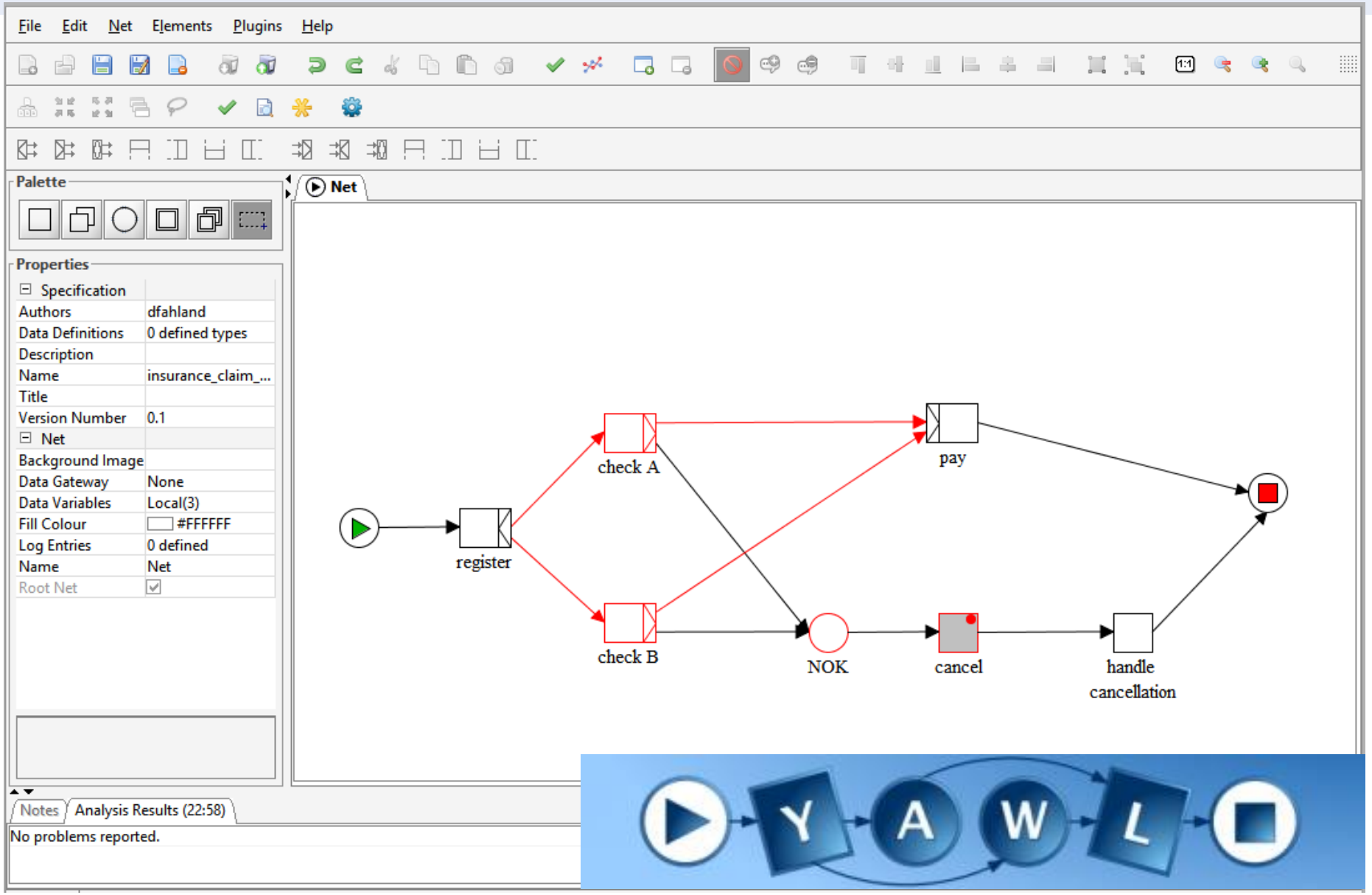
d.fahland@tue.nl / @dfahland

Describing, Discovering, and Understanding Multi-Dimensional Processes

TU/e
EINDHOVEN UNIVERSITY OF TECHNOLOGY

Computer Science / Process Analytics

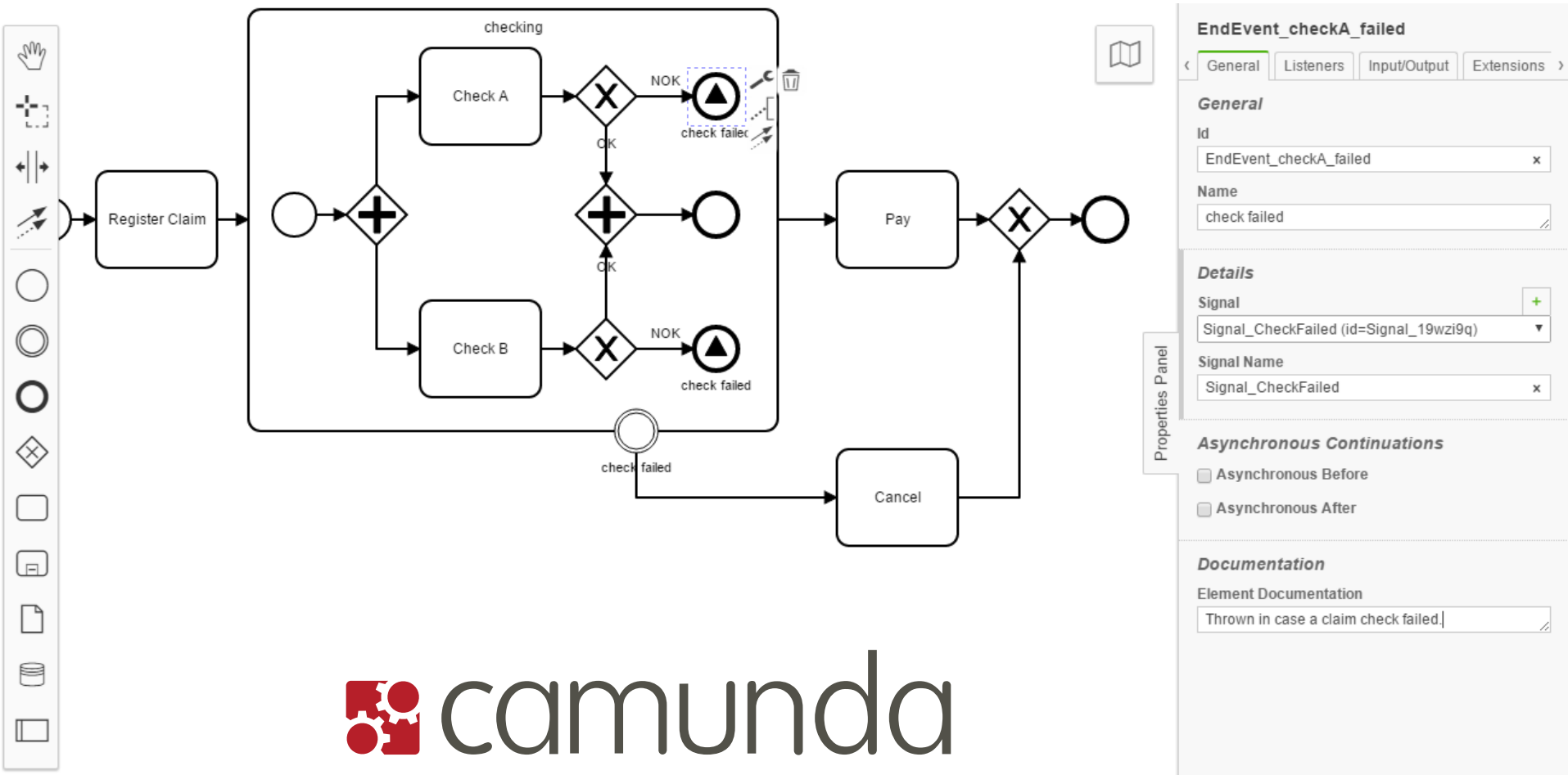
Processes and Petri Nets



Locality of transitions + places:

- Perfect fit for process modeling, execution, analysis

Processes and Petri Nets

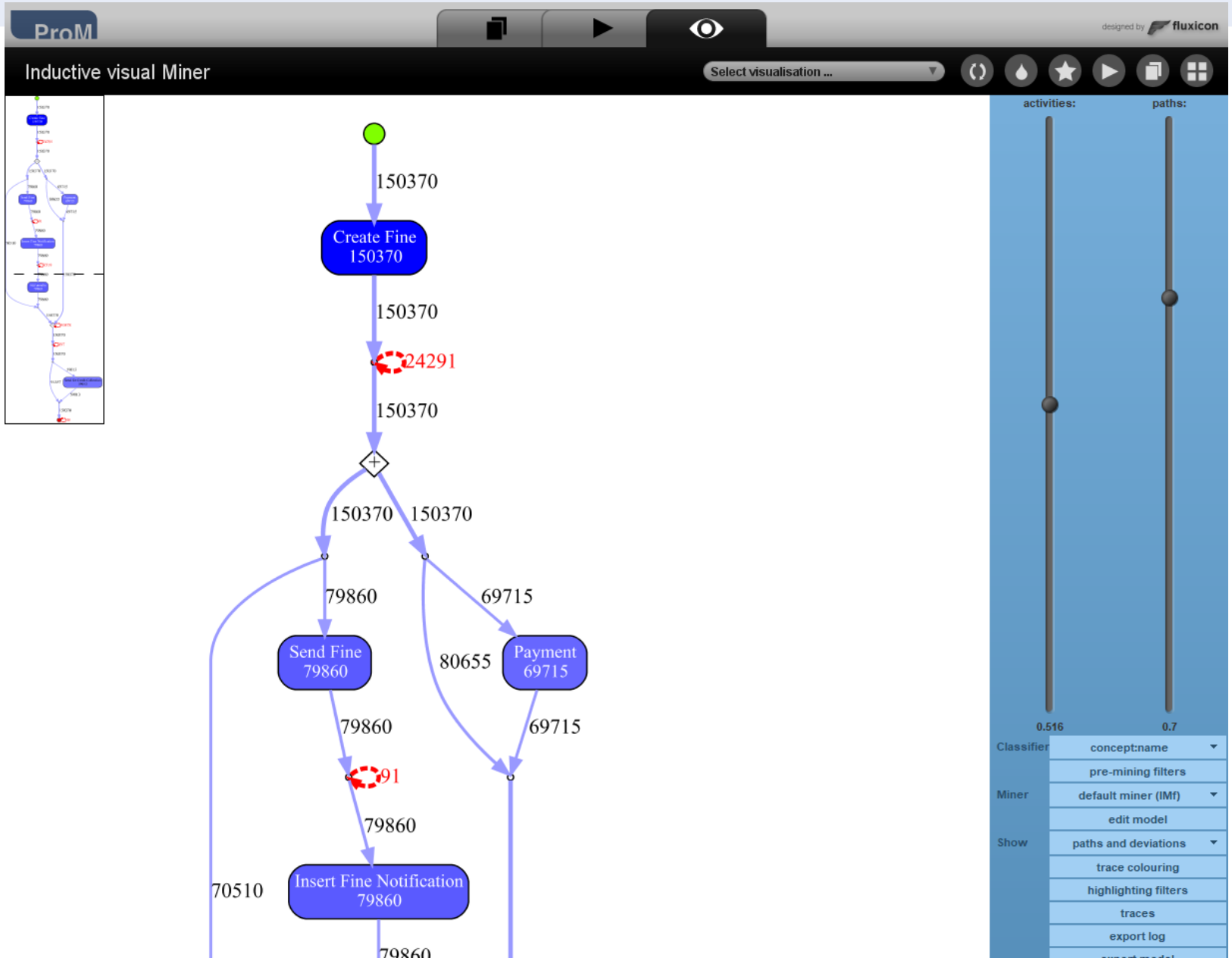


camunda

Locality of transitions + places:

- Perfect fit for process modeling, execution, analysis
- Heavily influenced design of industrial modeling languages, execution engines

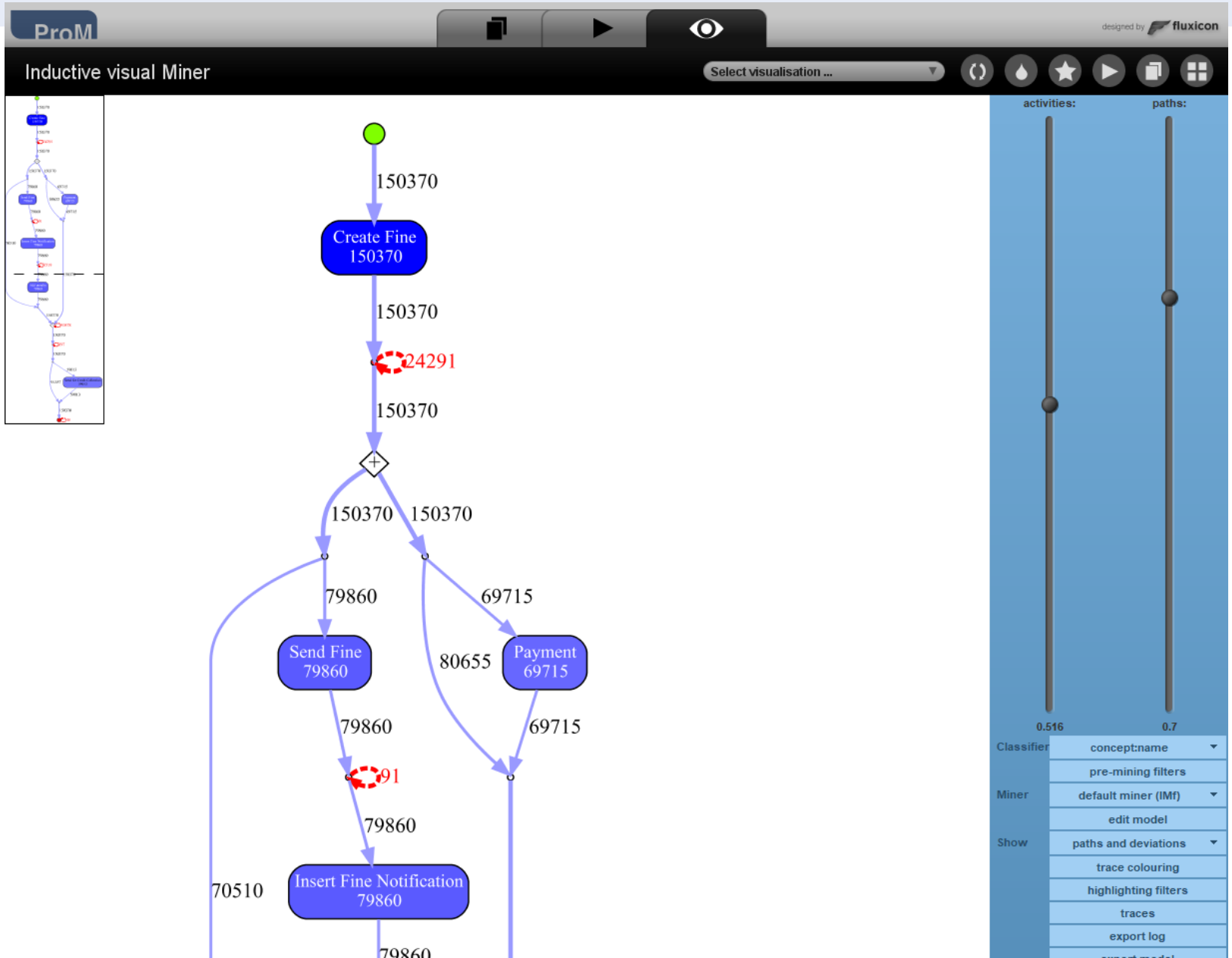
Processes and Petri Nets



Locality of transitions + places:

- Perfect fit for process modeling, execution, analysis
- Heavily influenced design of industrial process modeling languages, process execution engines
- Foundational to process mining

Processes and Petri Nets

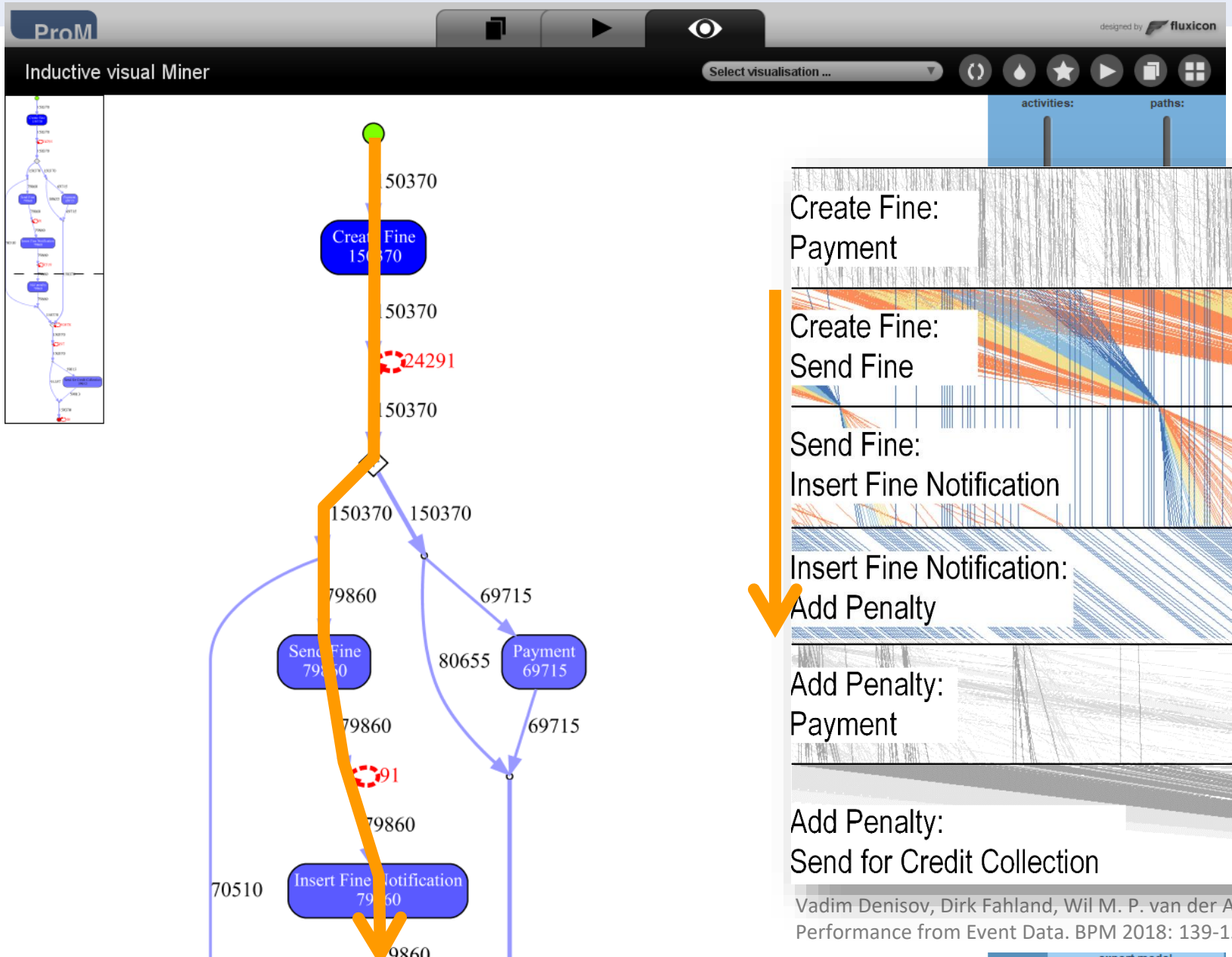


Locality of transitions + places:

- Perfect fit for process modeling, execution, analysis
- Heavily influenced design of industrial process modeling languages, process execution engines
- Foundational to process mining

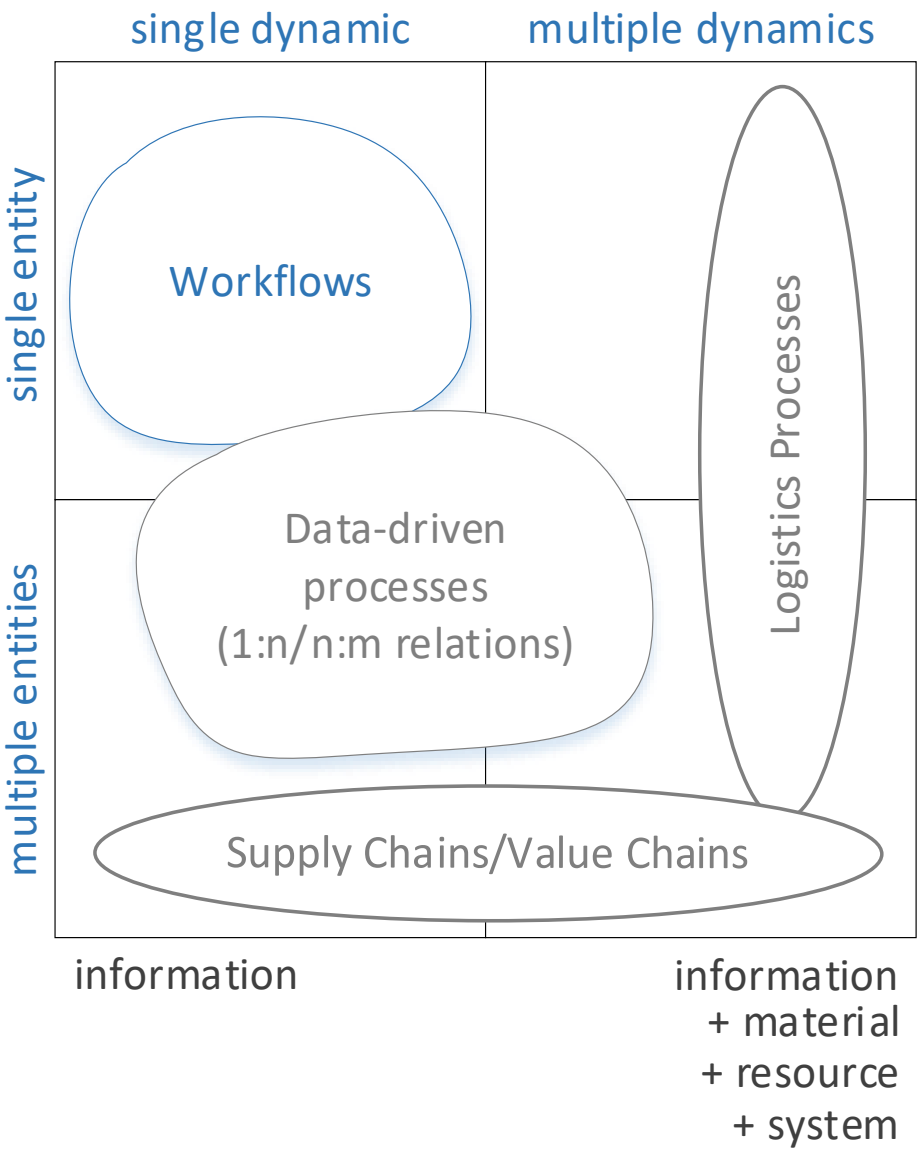
Assumption: one run = one process instance in isolation

Process instances are NEVER isolated

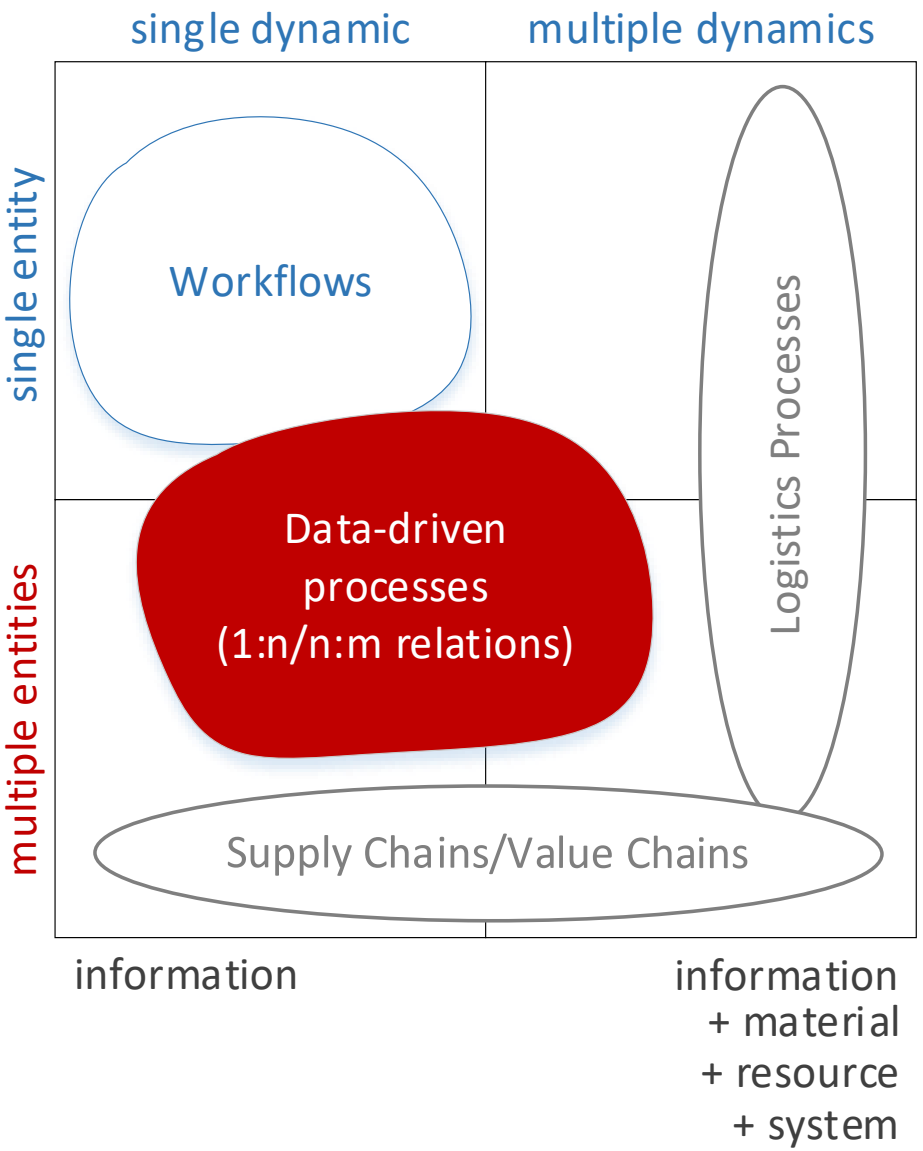


Vadim Denisov, Dirk Fahland, Wil M. P. van der Aalst: Unbiased, Fine-Grained Description of Processes Performance from Event Data. BPM 2018: 139-157

Dimensions in Processes

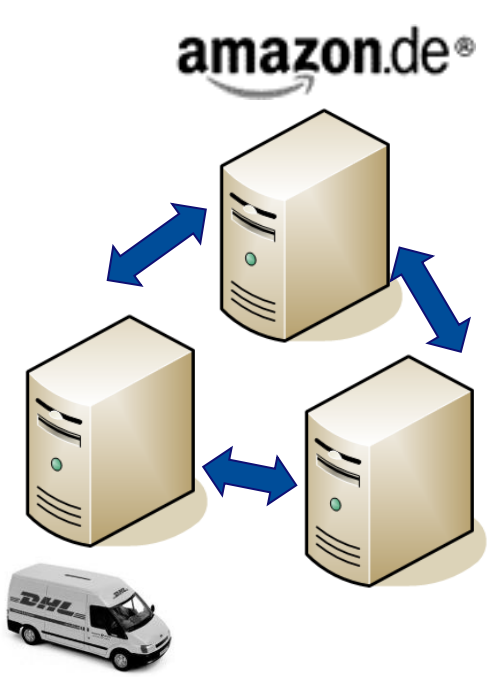


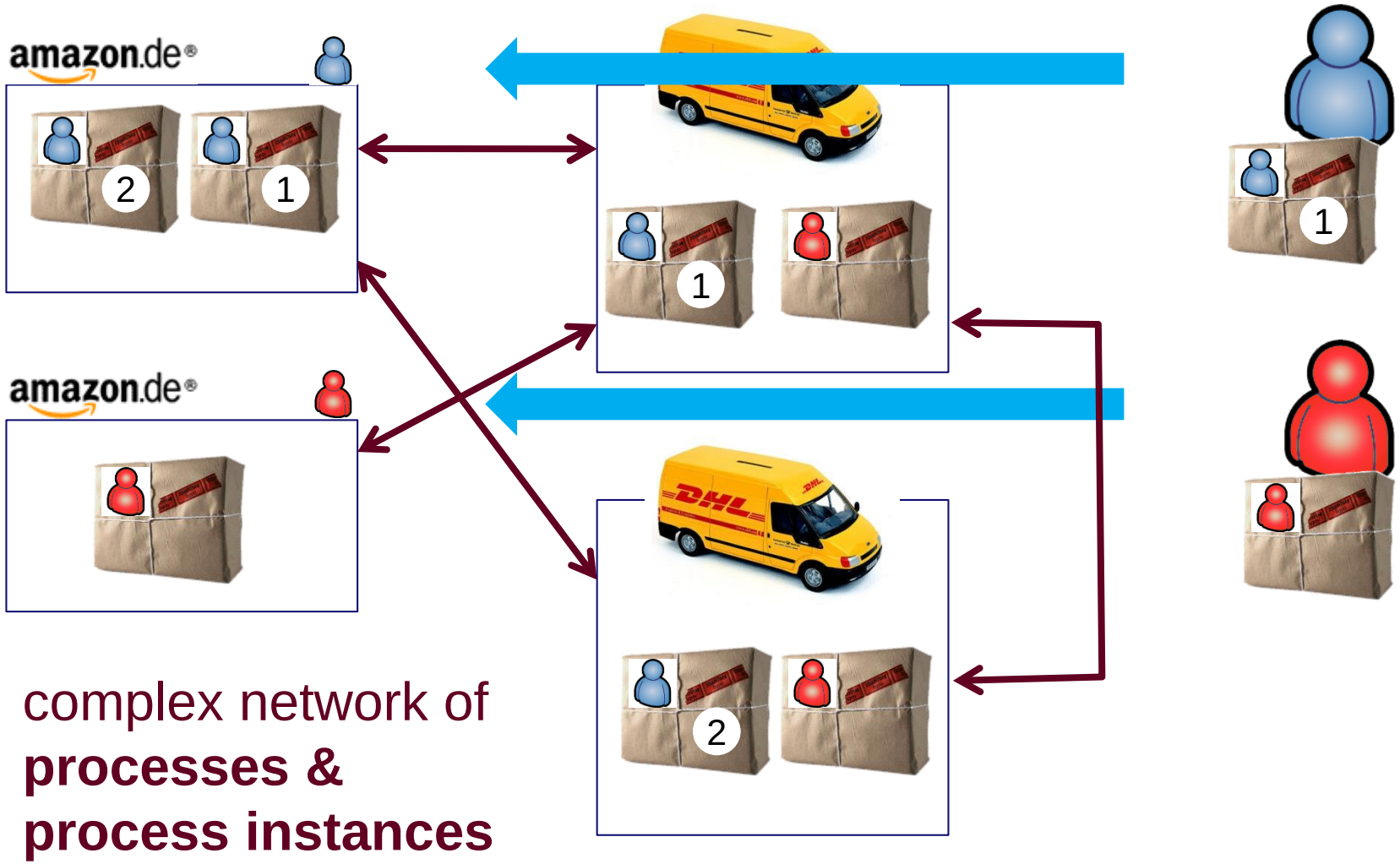
Multi-Dimensional Processes



- **Processes with multiple entities**
  - How do they look like in practice?
  - Where does current modeling fail?
    - Implementation
    - Process mining
  - A proposal
  - New research questions

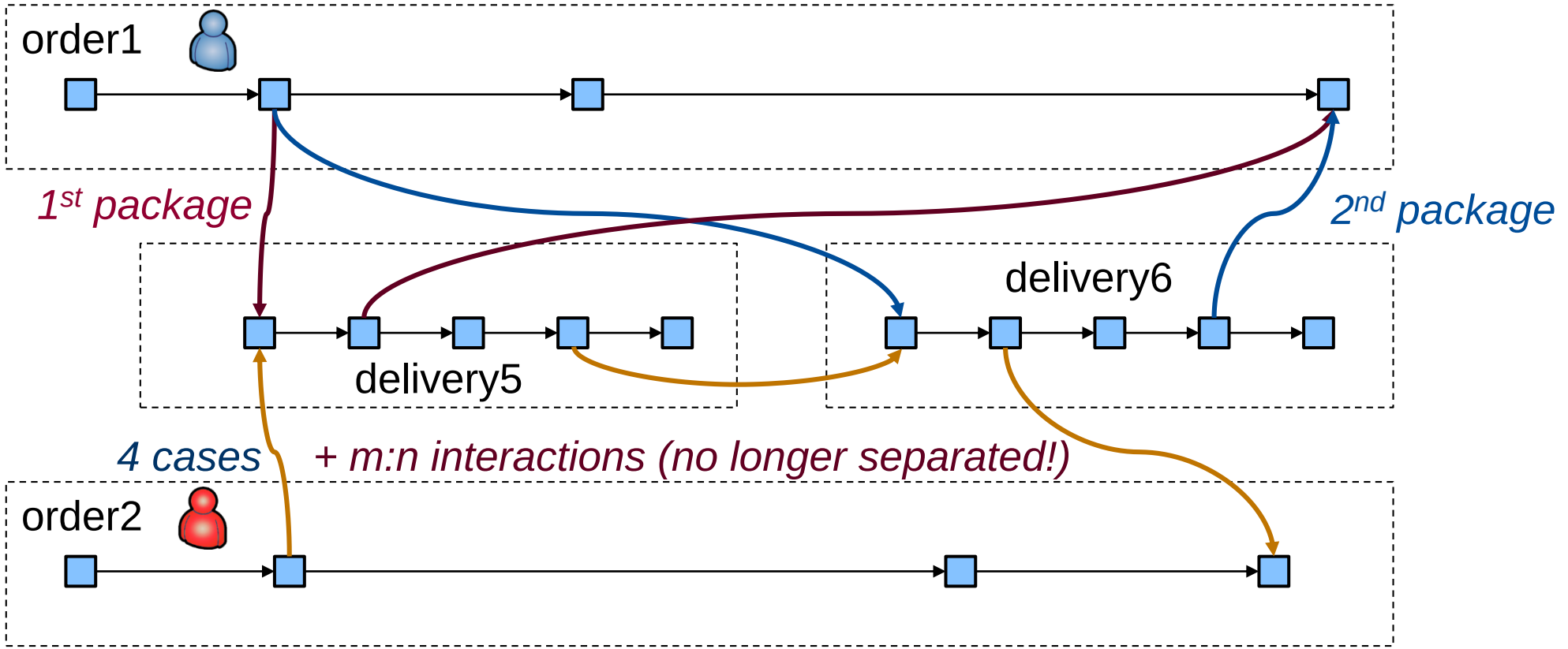
Multi-Dimensional Dynamics

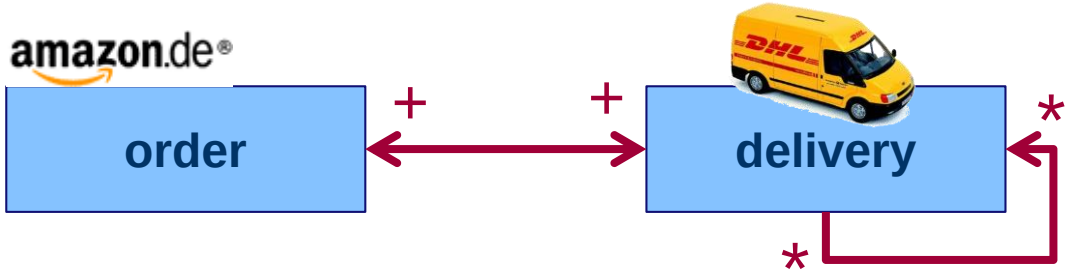




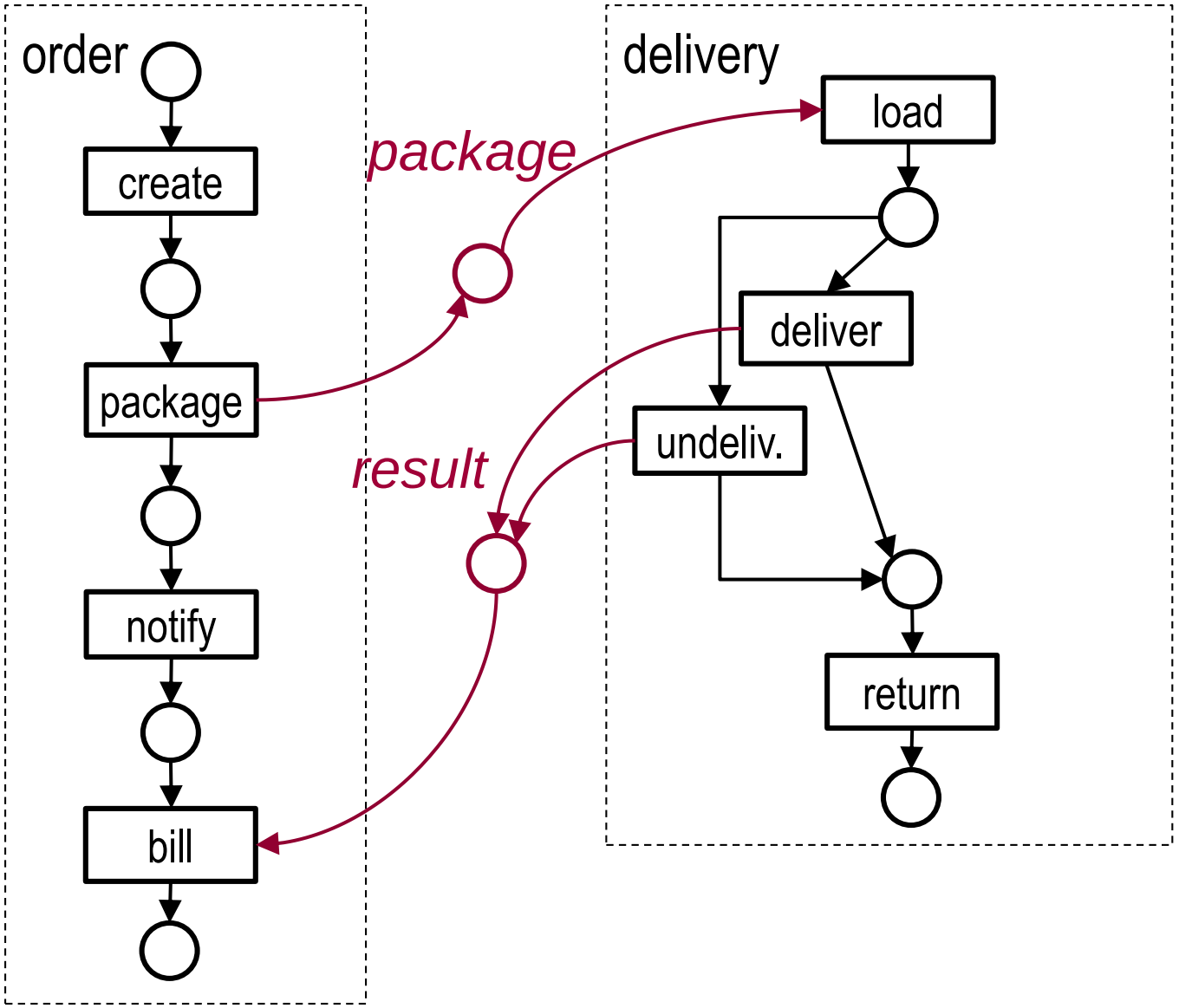
complex network of
processes &
process instances

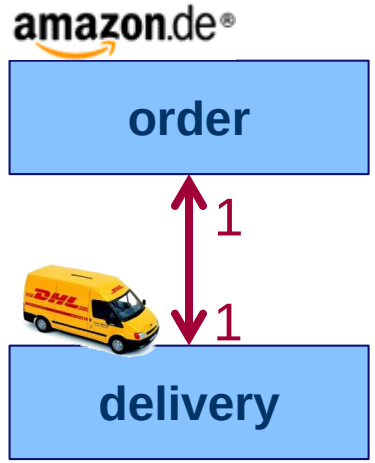
Interacting Cases - M:N



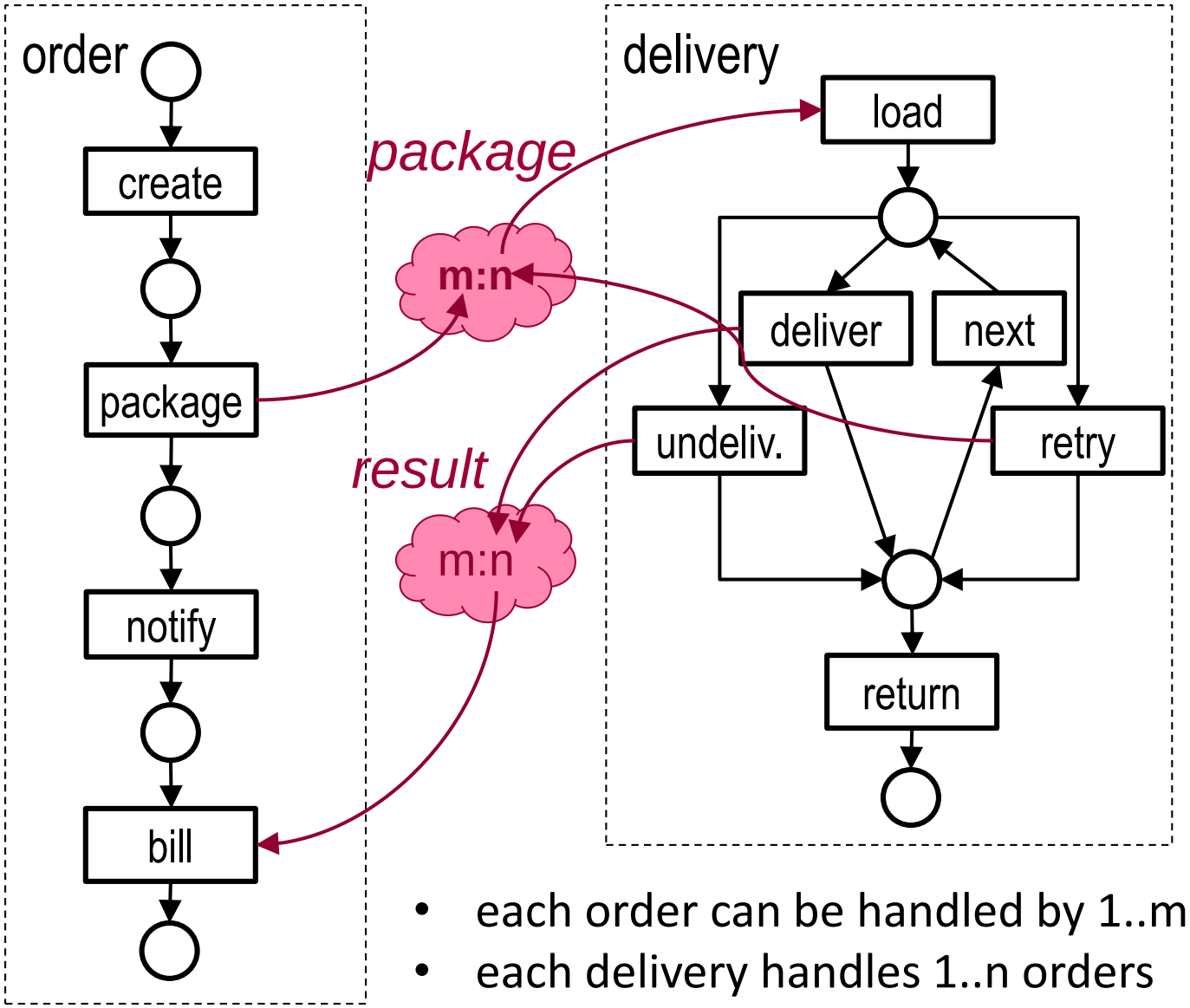


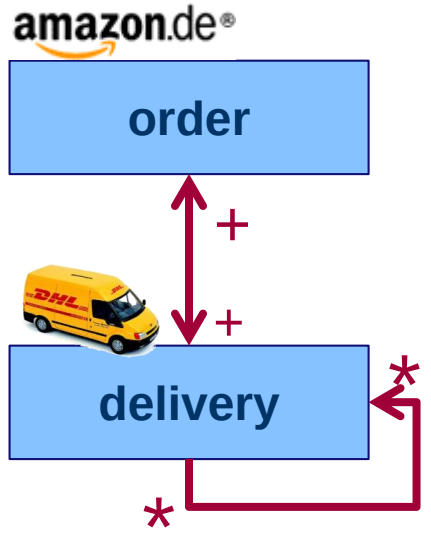
Modeling 1:1 Interactions





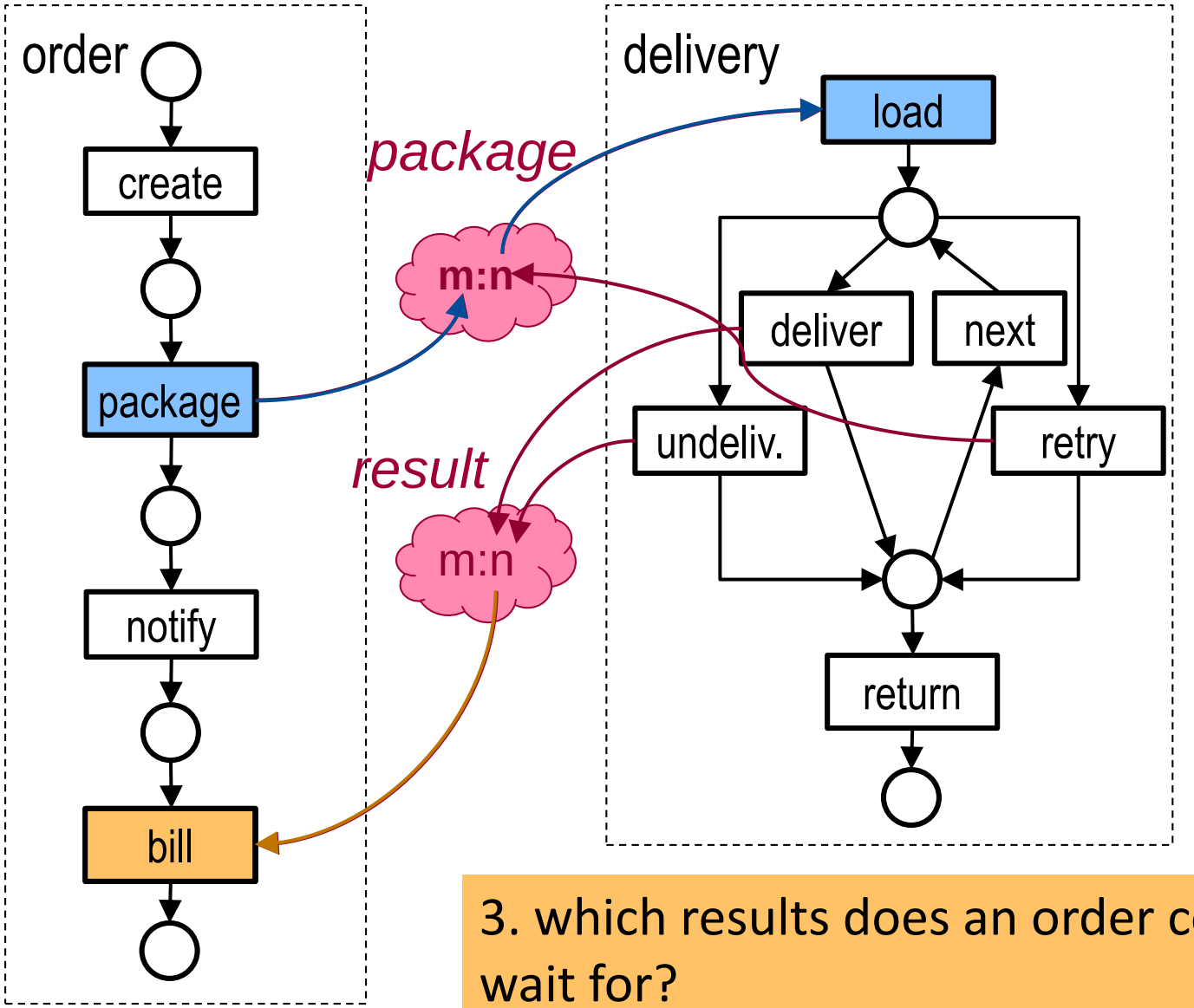
Modeling M:N Interactions?





- each order can be handled by 1..m deliveries
- each delivery handles 1..n orders

What is missing to have an executable model?

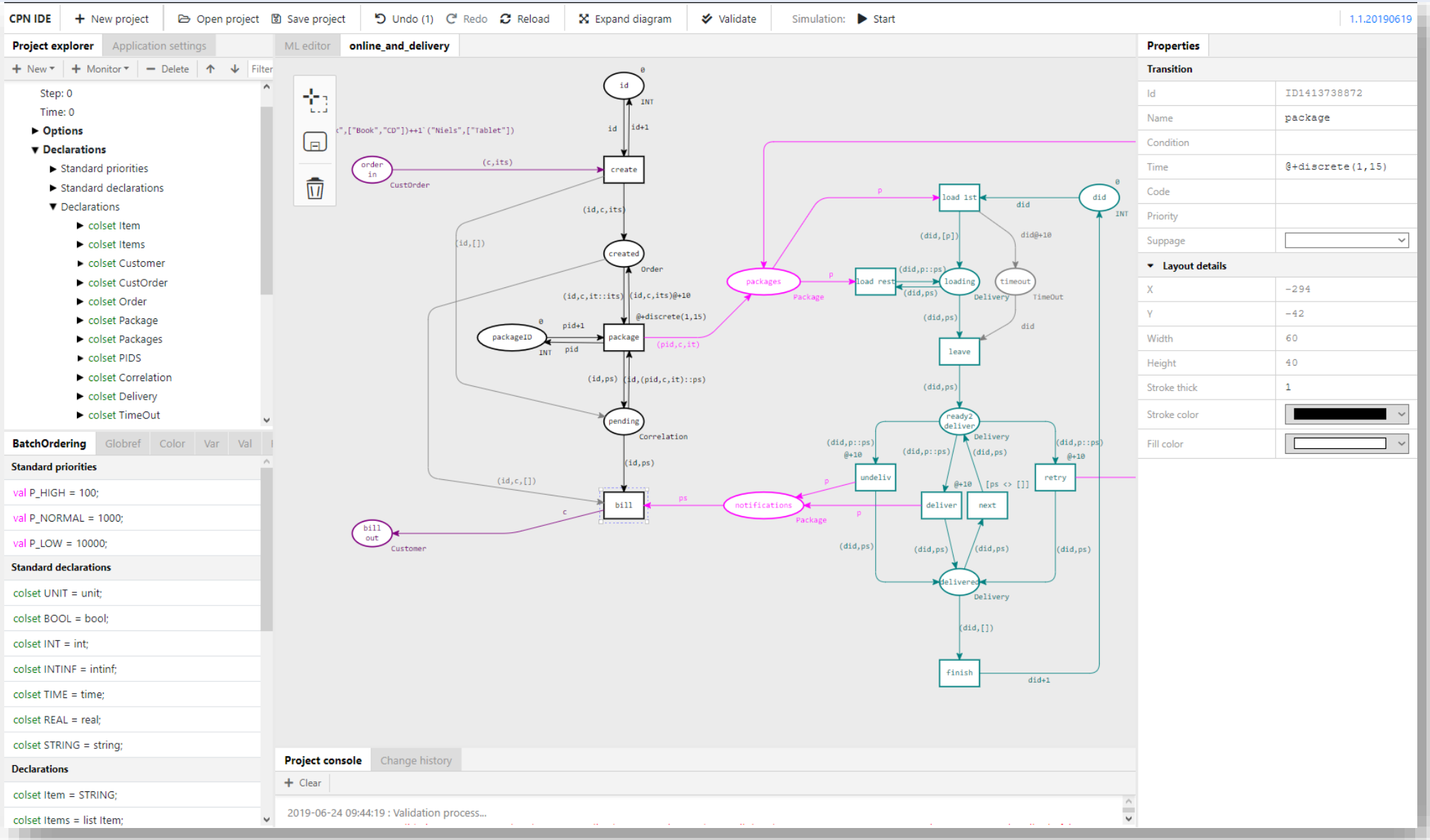


1. distinguish different packages

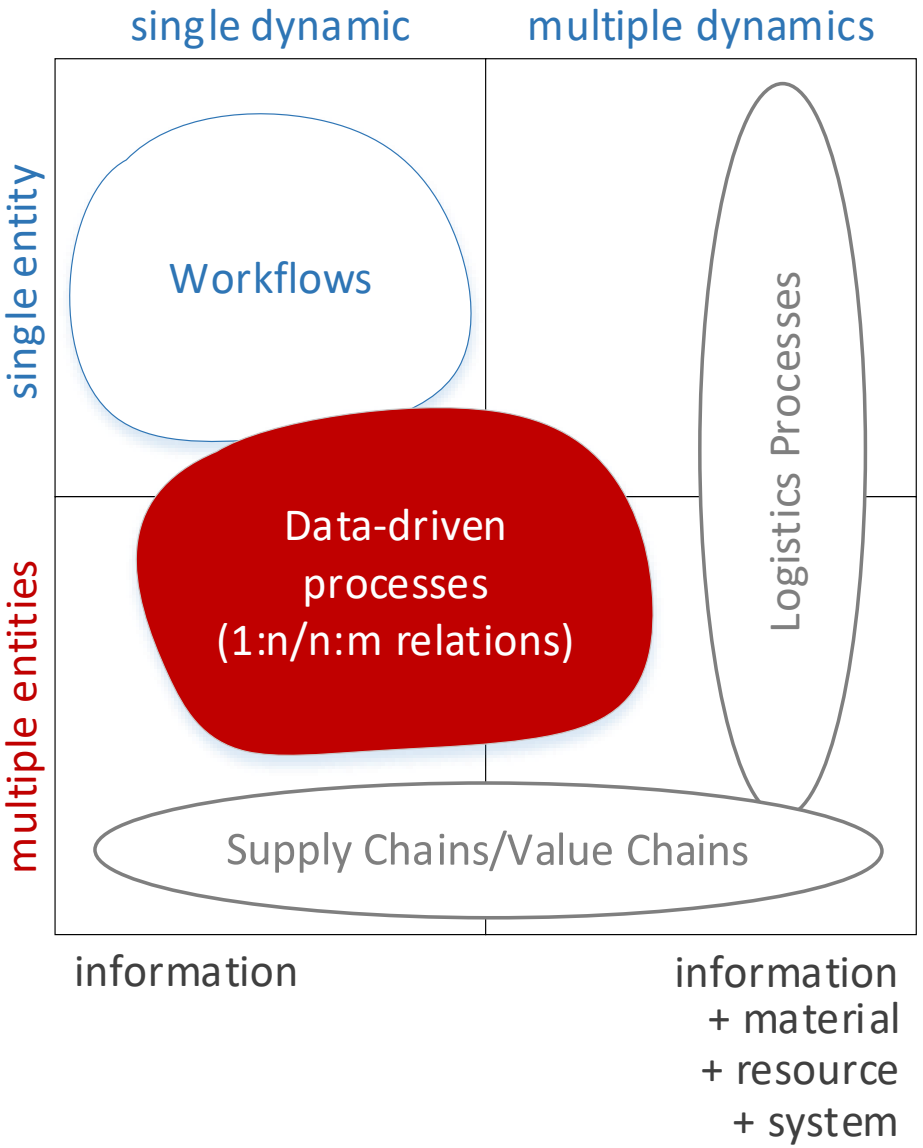
2. how many packages and to which delivery does a package go?

3. which results does an order consume/ wait for?

Can be modeled with Coloured Petri Nets

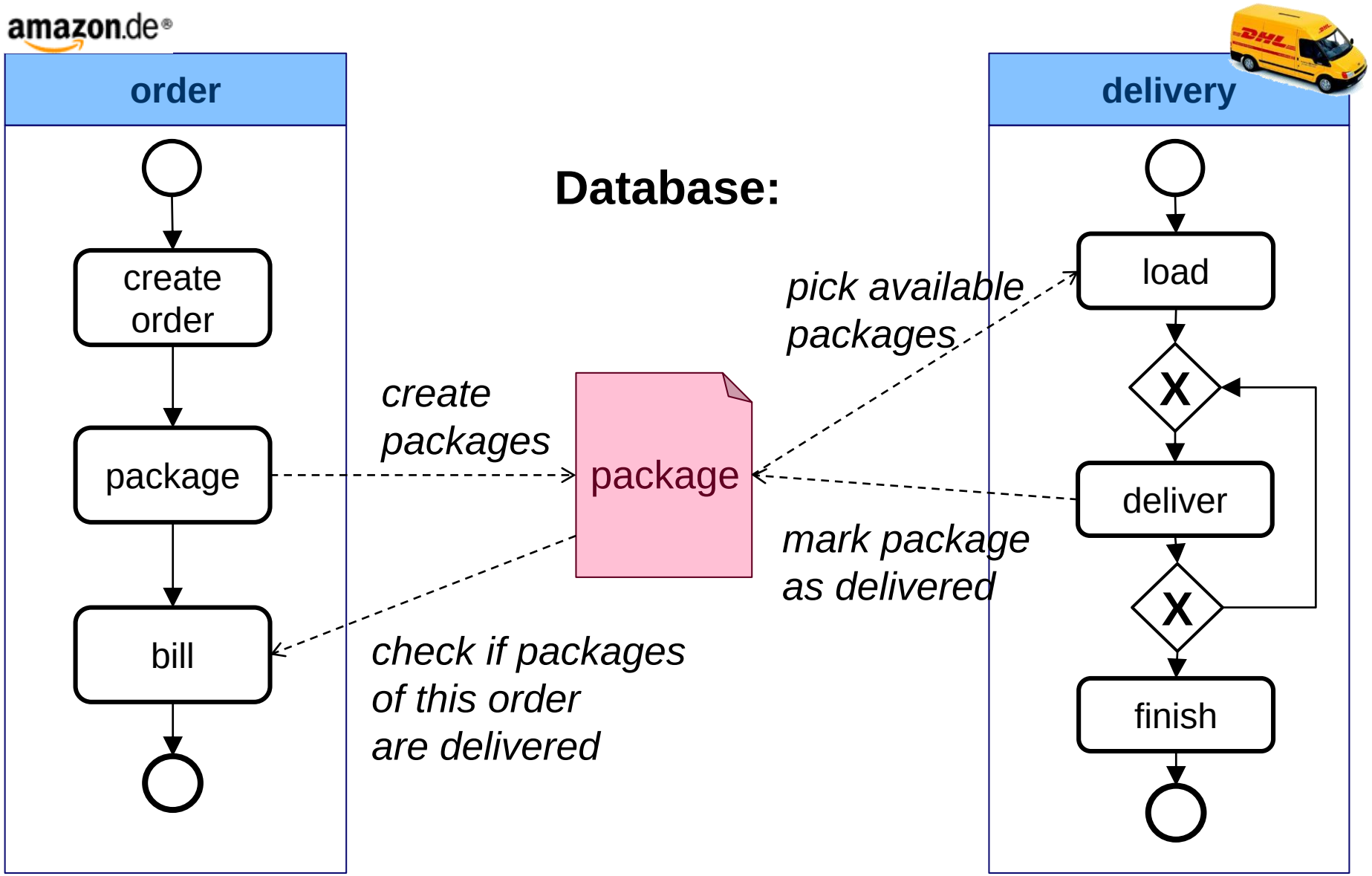


Multi-Dimensional Processes



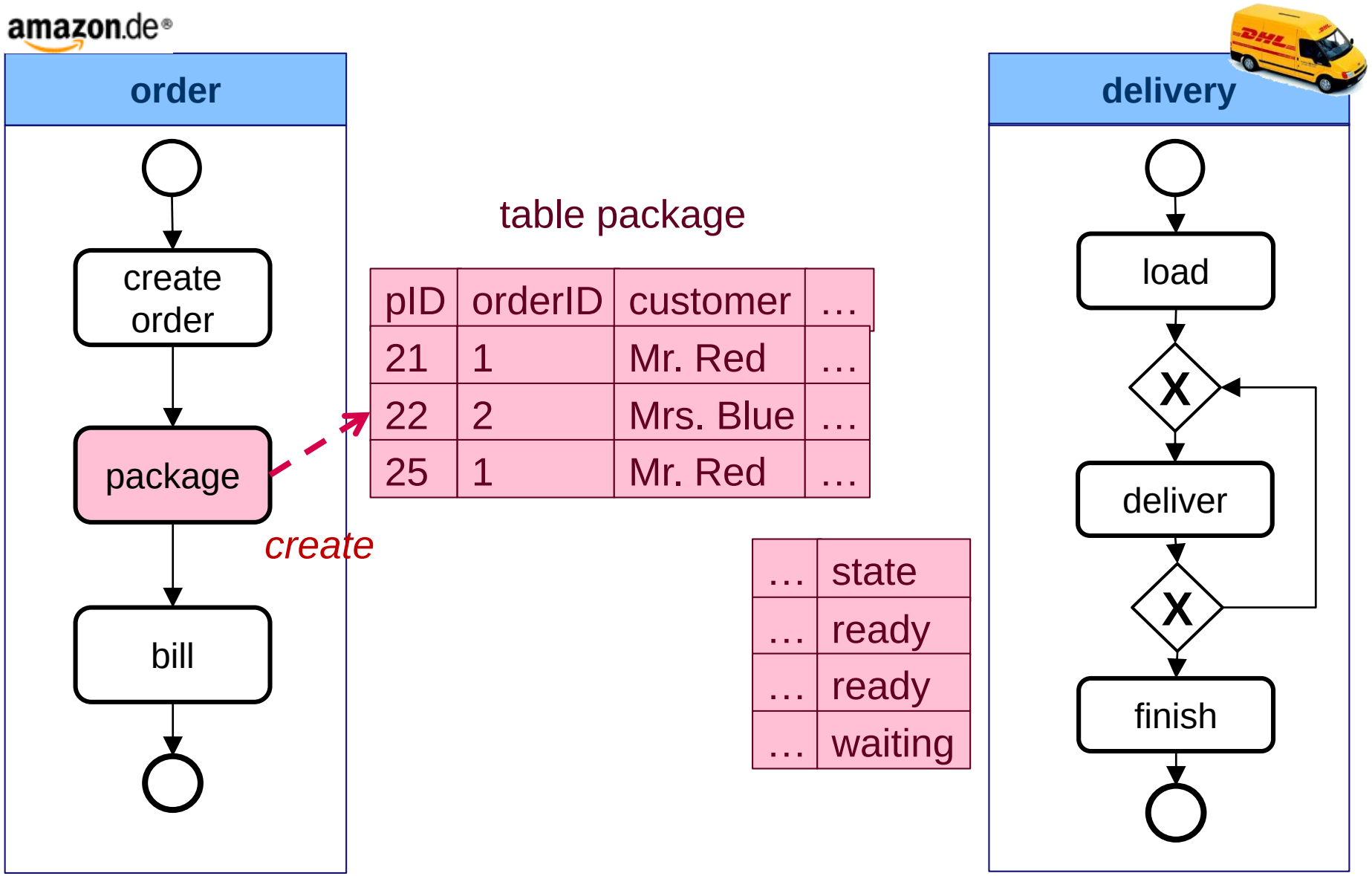
- **Processes with multiple entities**
  - How do they look like in practice?
  - **Where does current modeling fail?**
    - **Implementation**
    - Process mining
  - A proposal
  - New research questions

Implementation: BPMN + Relational Data + Operations



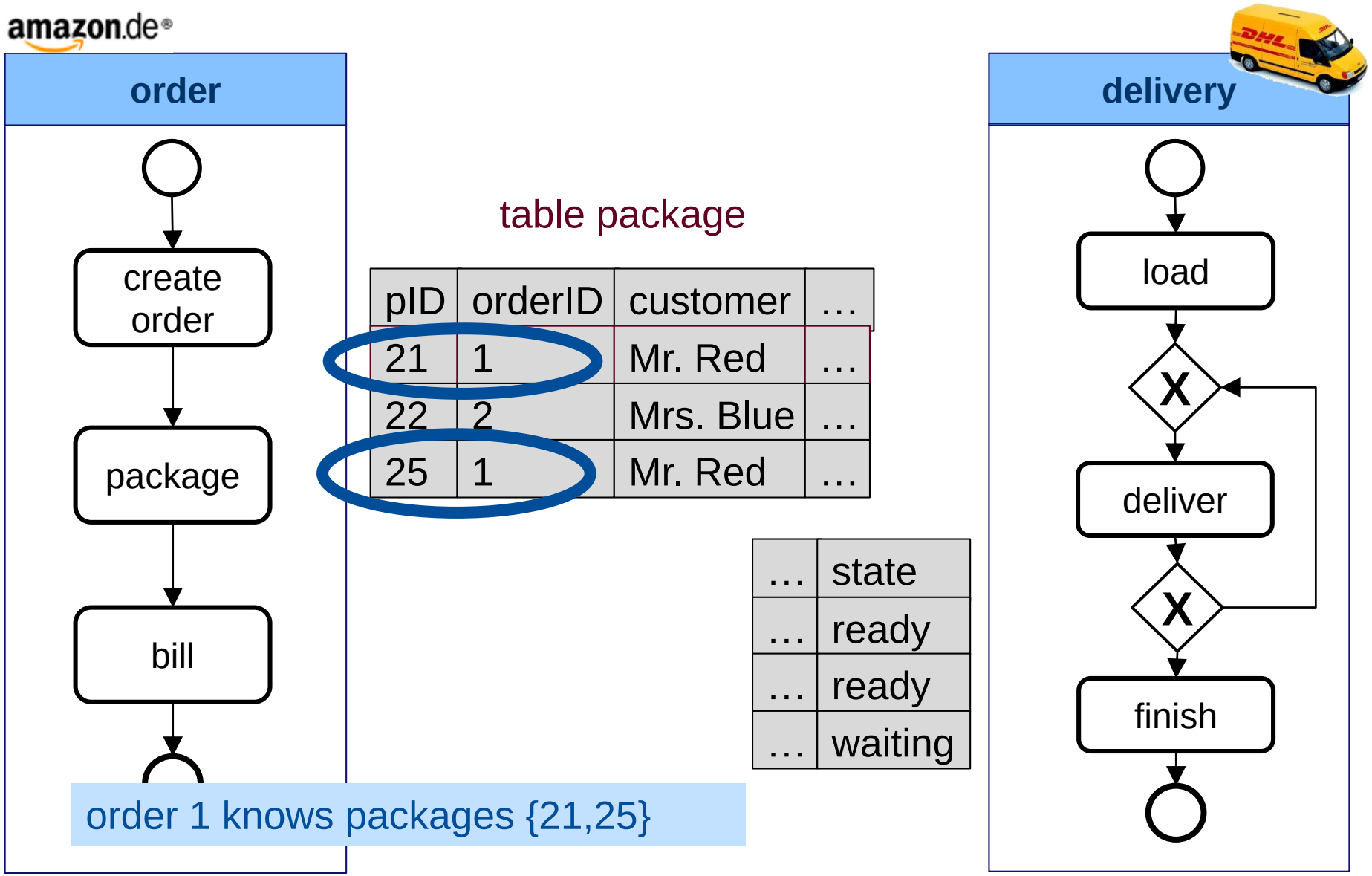
BPMN + Relational Data + Operations

create packages



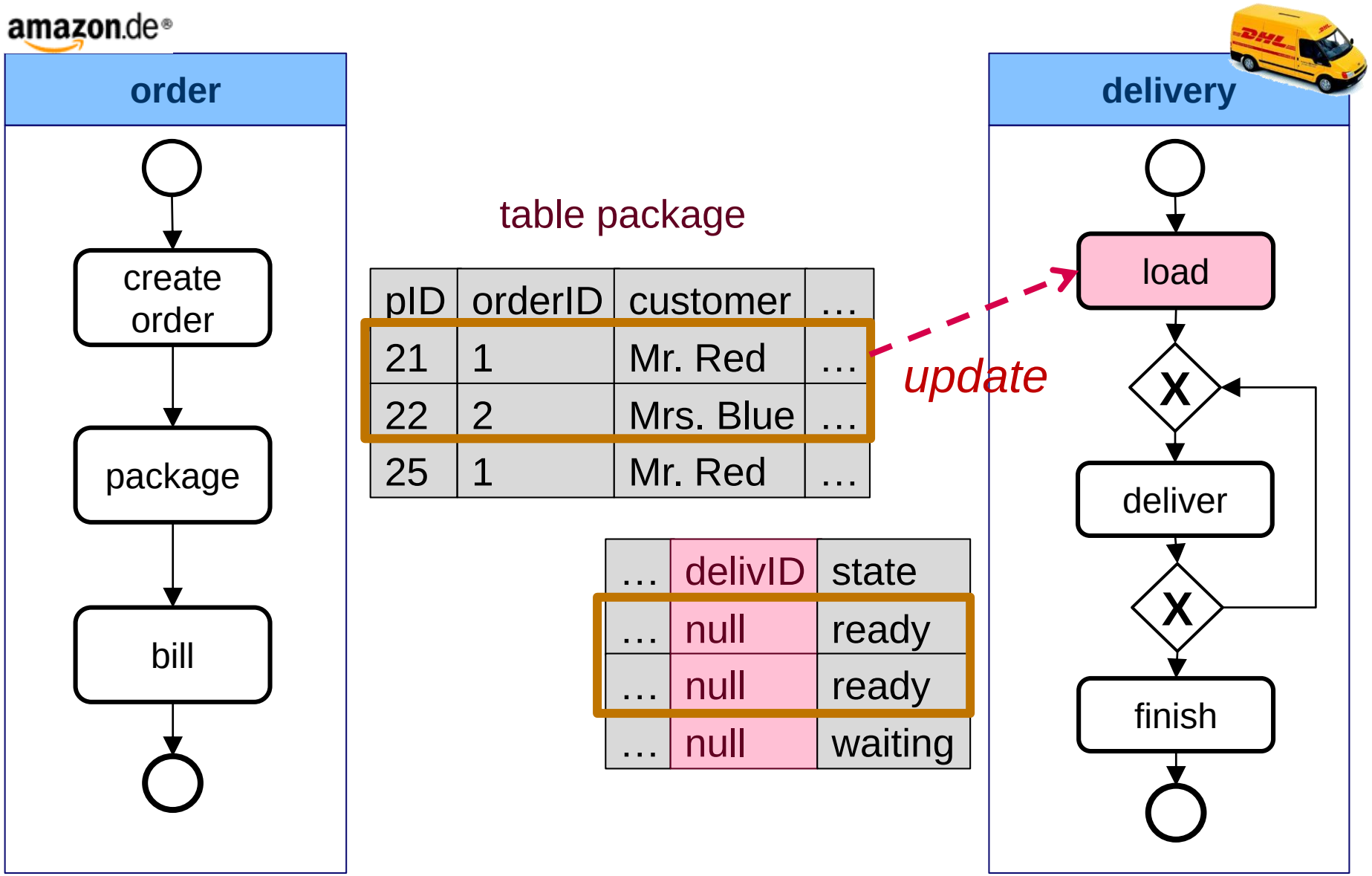
BPMN + Relational Data + Operations

… attributes referring to process instances



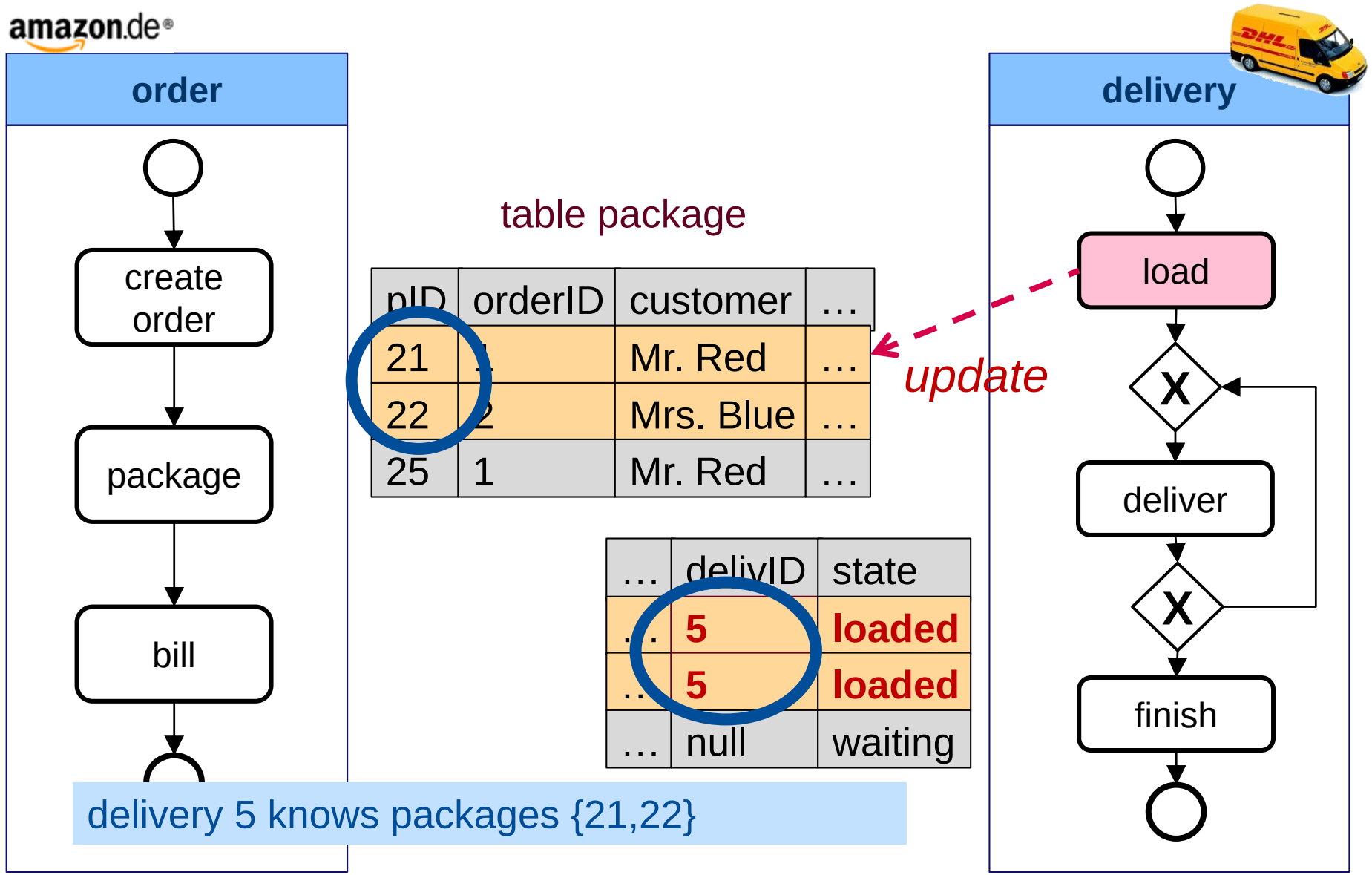
BPMN + Relational Data + Operations

associate ready packages to "delivery 5"



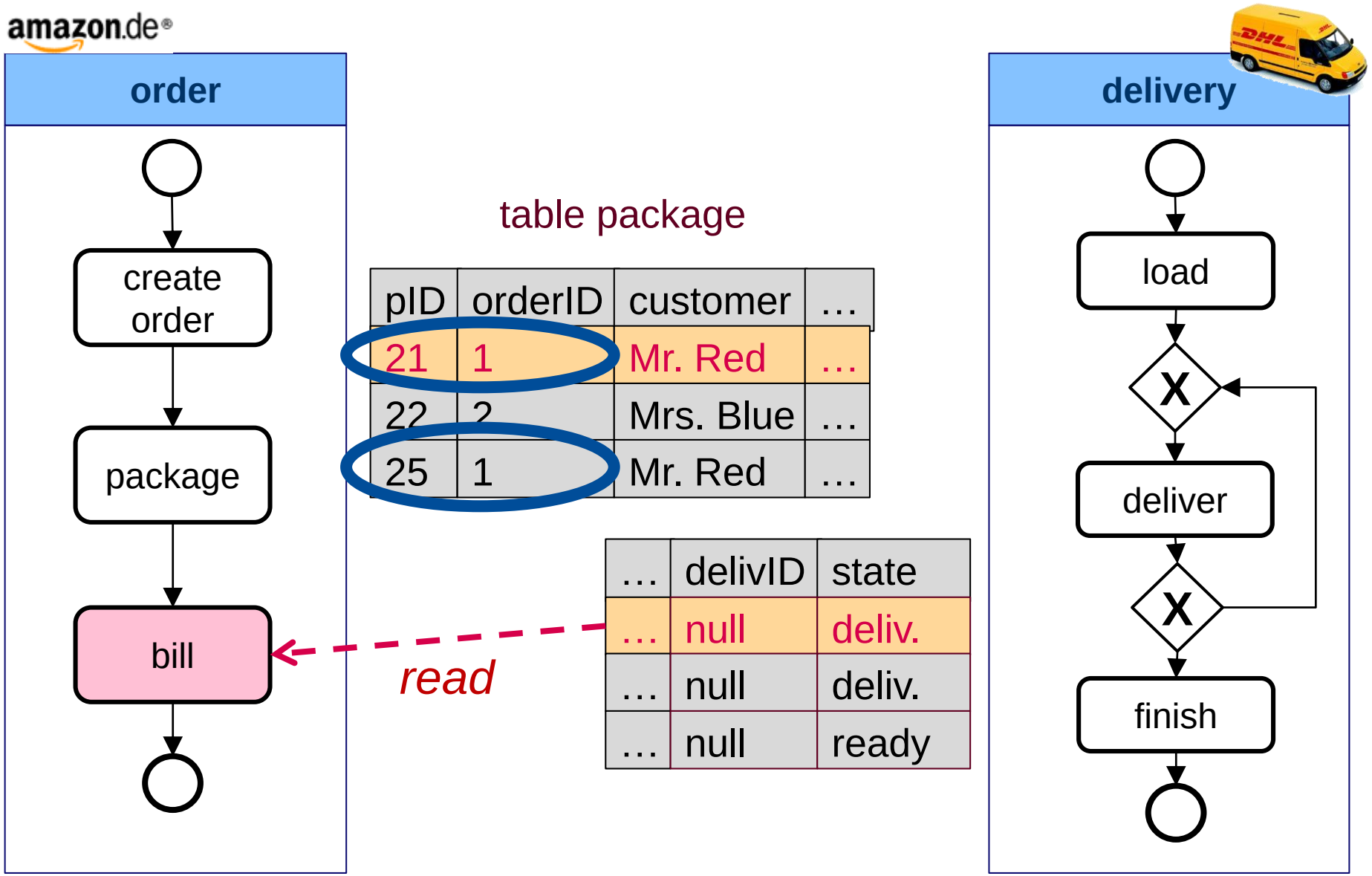
BPMN + Relational Data + Operations

associate ready packages to "delivery 5"

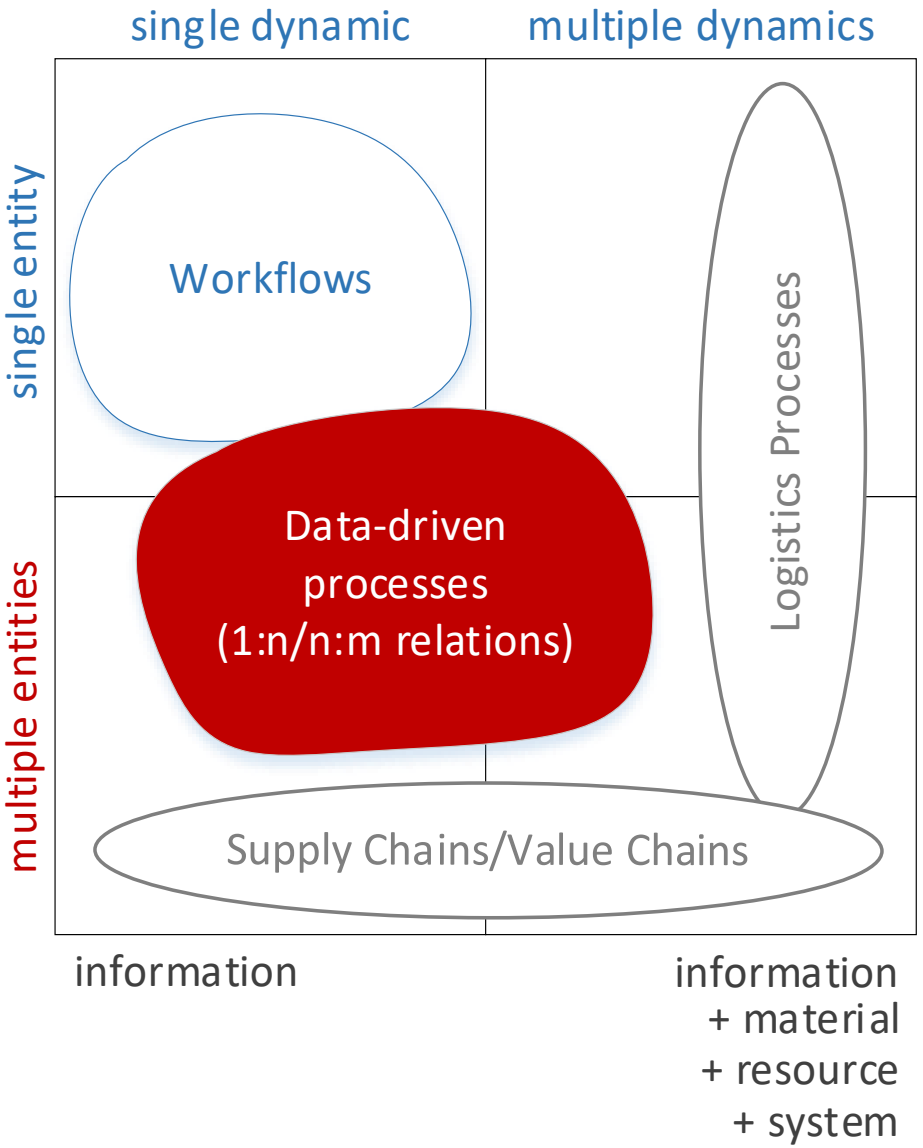


BPMN + Relational Data + Operations

complex sync: bill only when all delivered

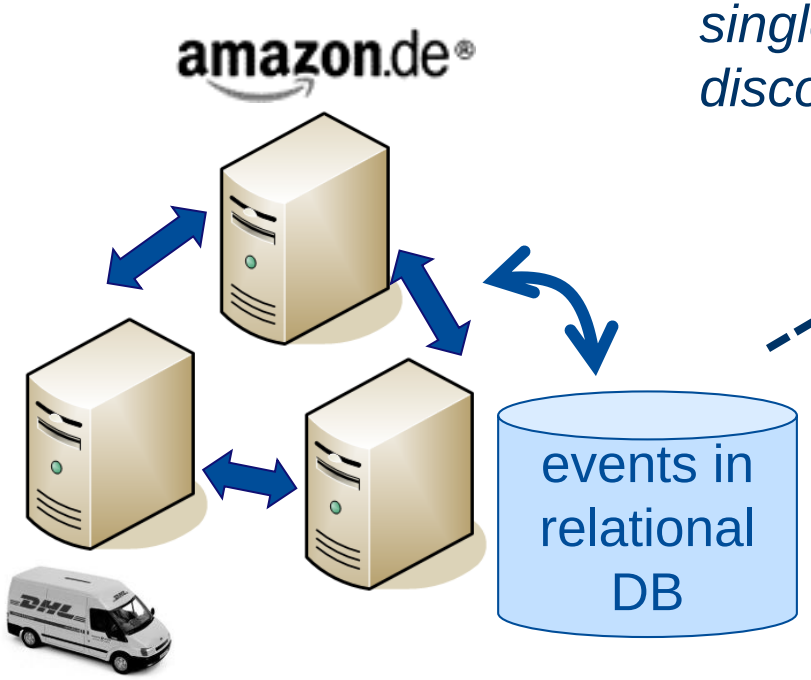


Multi-Dimensional Processes

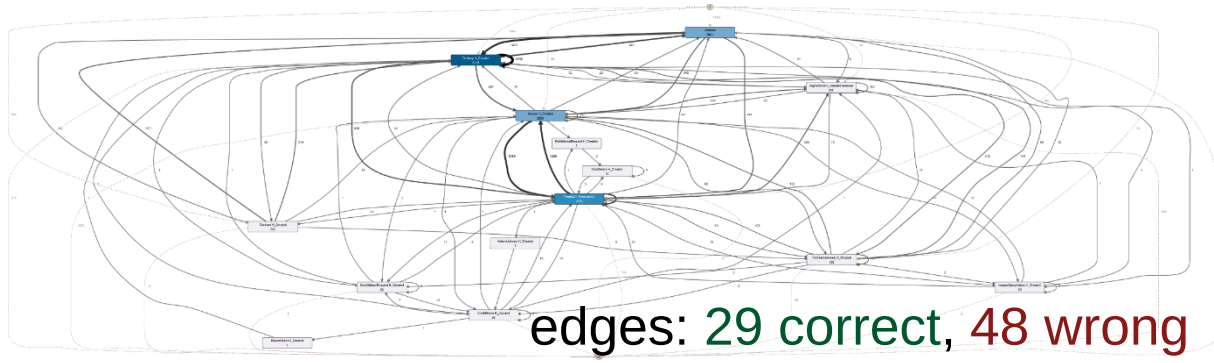


- **Processes with multiple entities**
  - How do they look like in practice?
  - Where does current modeling fail?
    - Implementation
    - **Process mining**
  - A proposal
  - New research questions

Process Mining on Multi-Dimensional Data



single-dimensional discovery

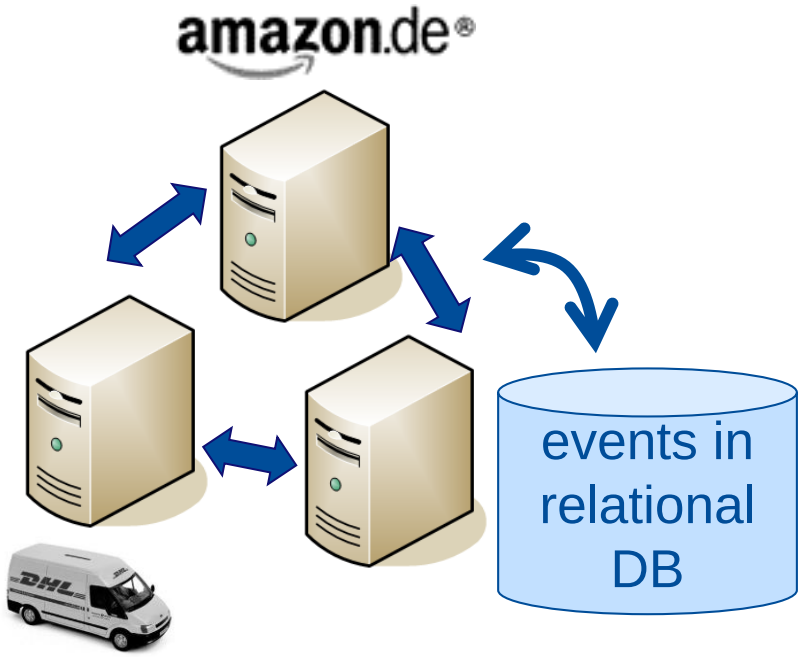


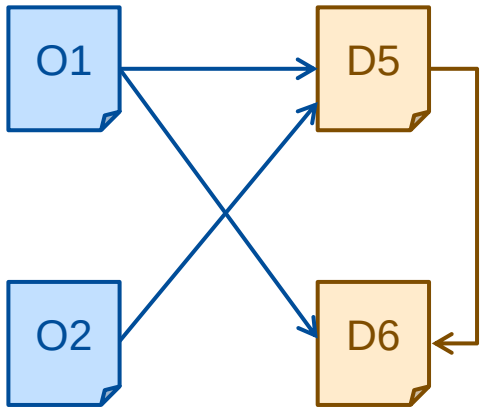
edges: 29 correct, 48 wrong

2 months SAP data

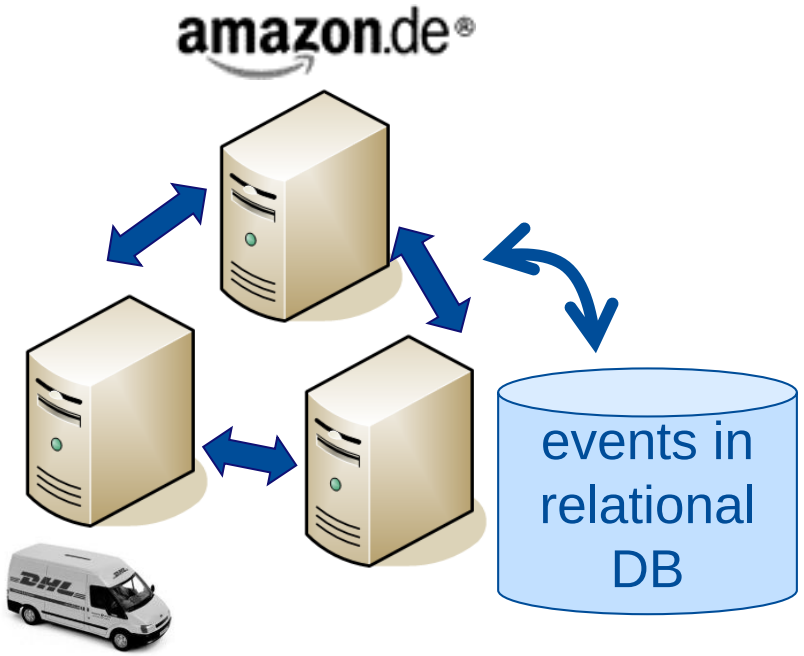
E. Nooijen, B. v. Dongen, and D. Fahland, "**Automatic Discovery of Data-Centric and Artifact-Centric Processes**," in BPM Workshops. Springer, 2013, pp. 316–327.
Viara Popova, Dirk Fahland, Marlon Dumas: "**Artifact Lifecycle Discovery.**" Int. J. Cooperative Inf. Syst. 24(1) (2015)
Xixi Lu, Marijn Nagelkerke, Dirk Fahland: "**Discovering Interacting Artifacts from ERP systems**" IEEE Trans. On Services Computing DOI: 10.1109/TSC.2015.2474358

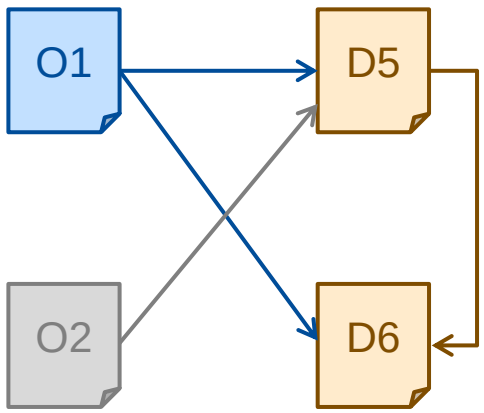
Why classical Process Mining (single case ID) fails





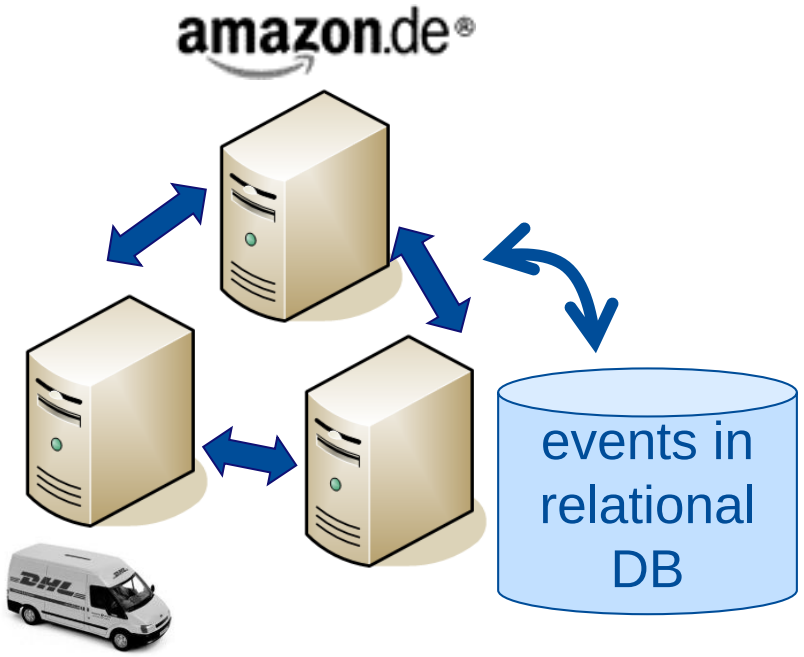
Why classical Process Mining (single case ID) fails

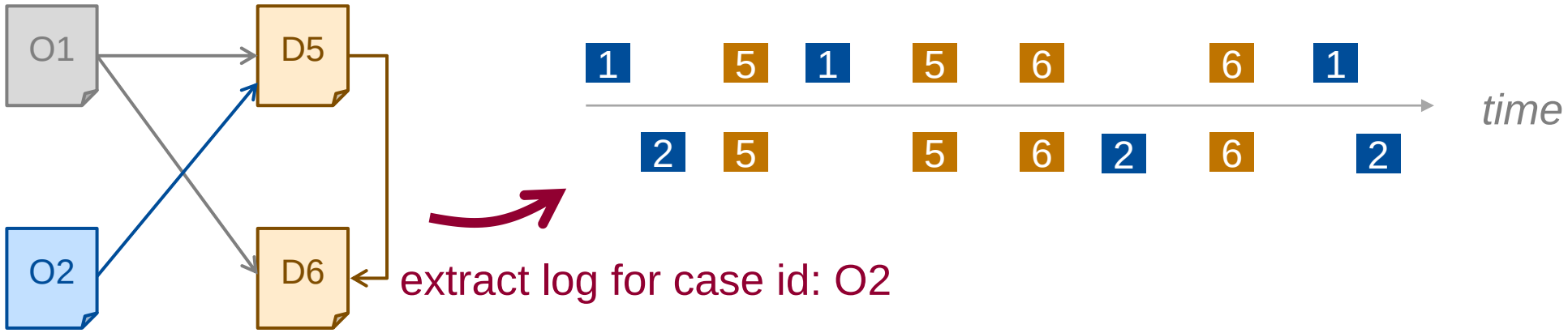




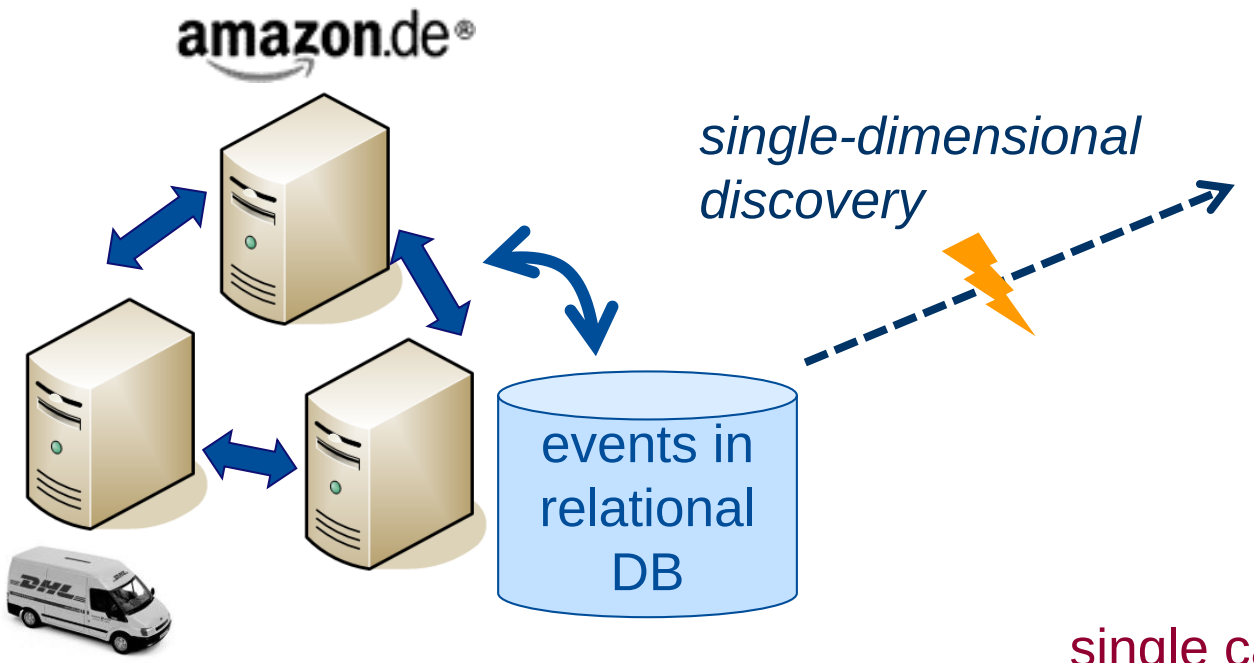
extract log for case id: O1

Why classical Process Mining (single case ID) fails

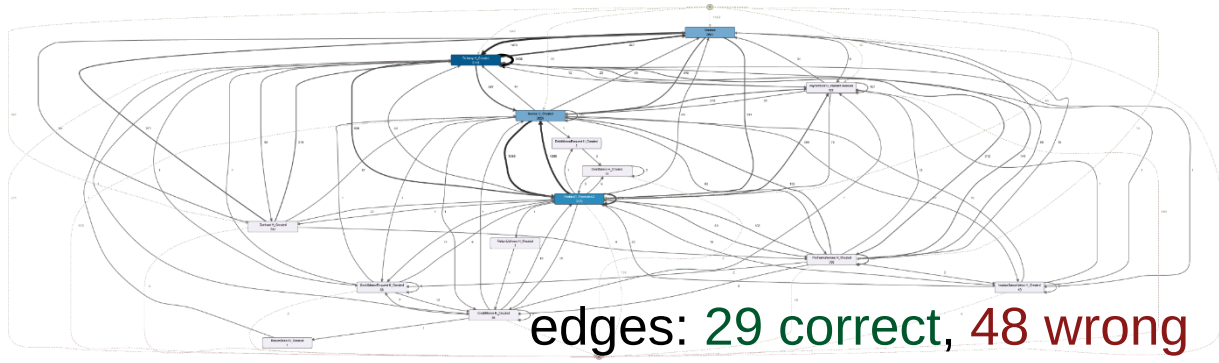




Why classical Process Mining (single case ID) fails

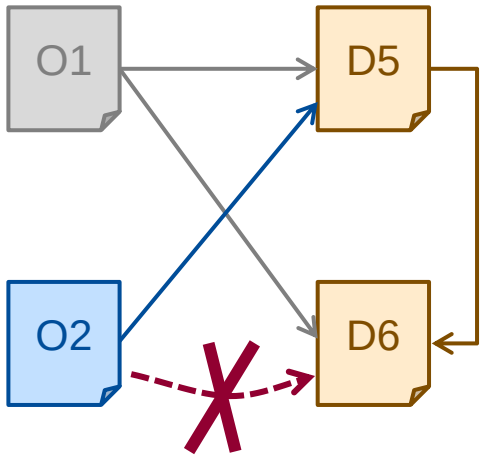


single-dimensional discovery

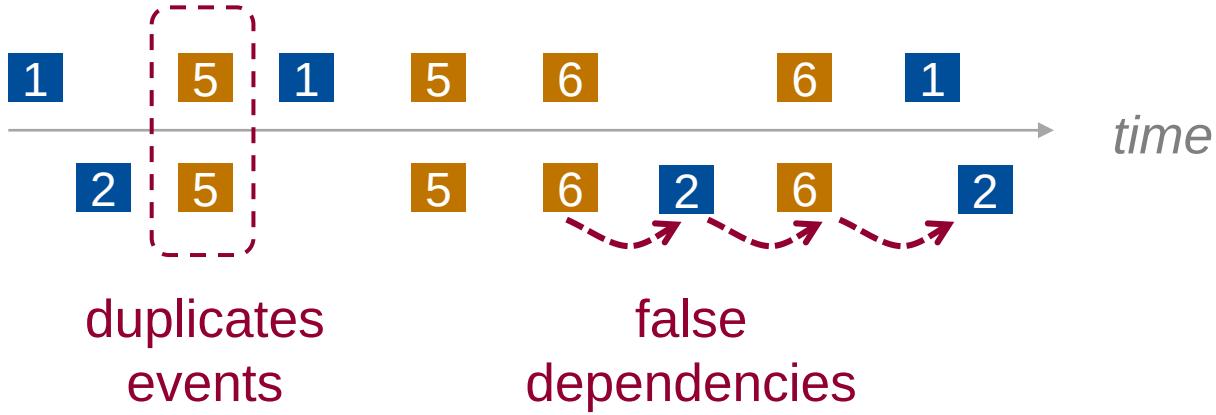


edges: 29 correct, 48 wrong

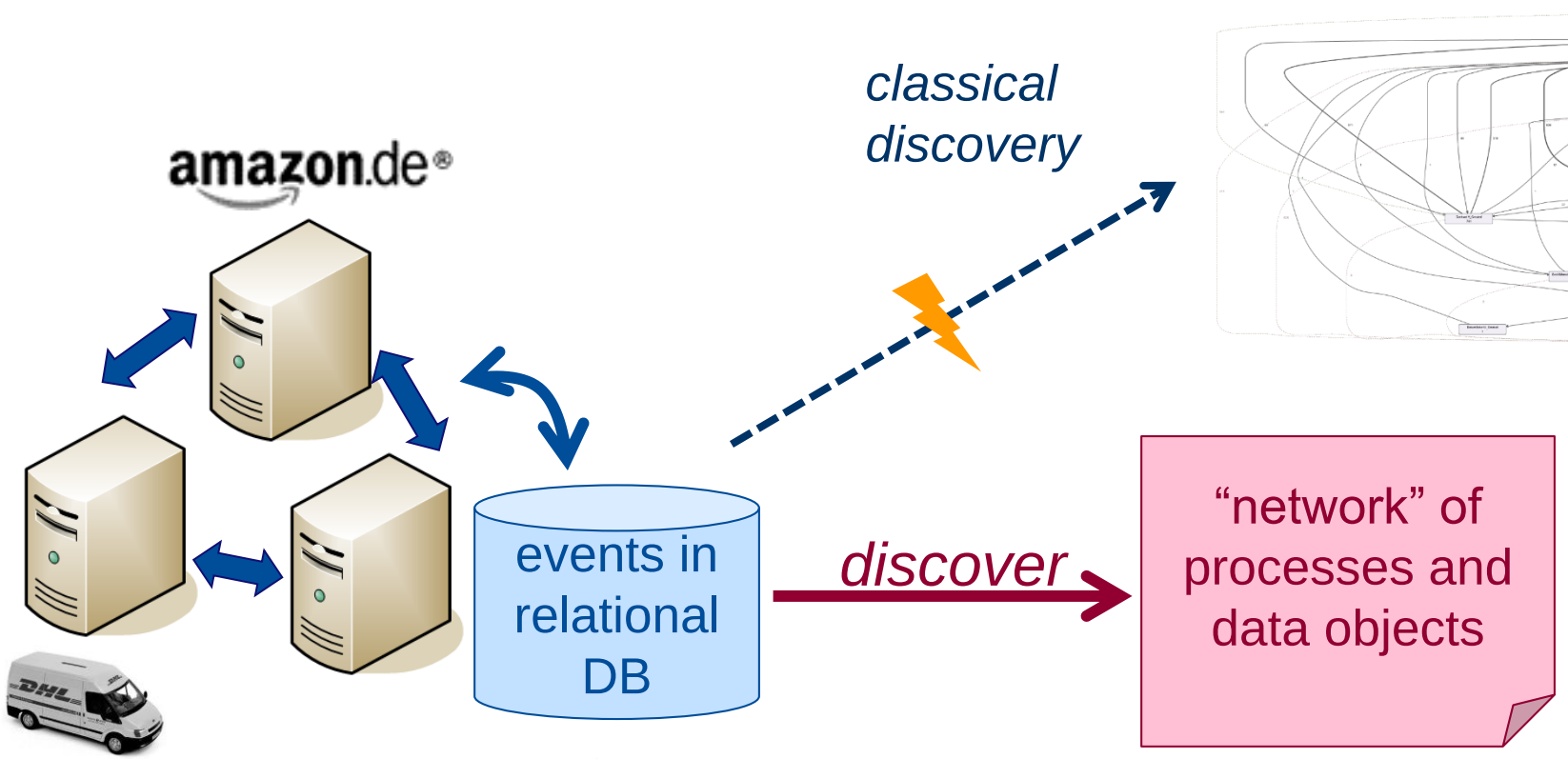
2 months SAP data

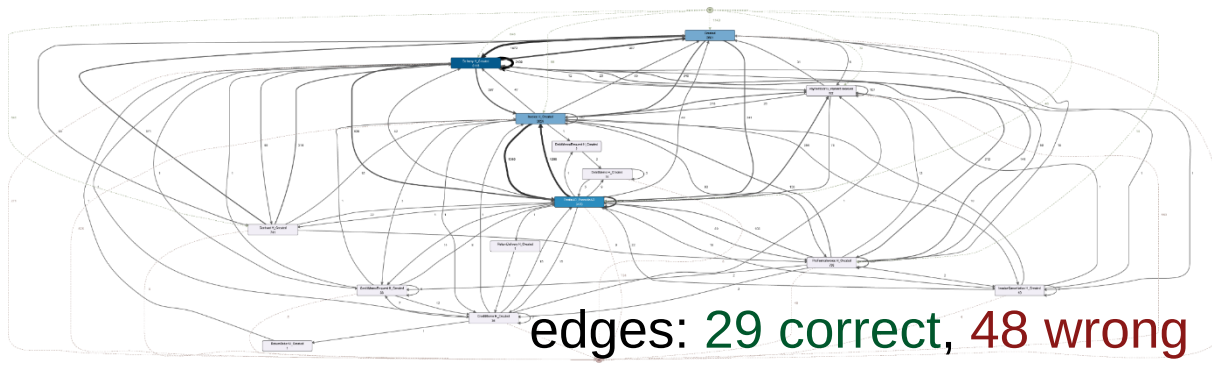


single case id → de-normalizes underlying data-structure



“Network” Process Mining



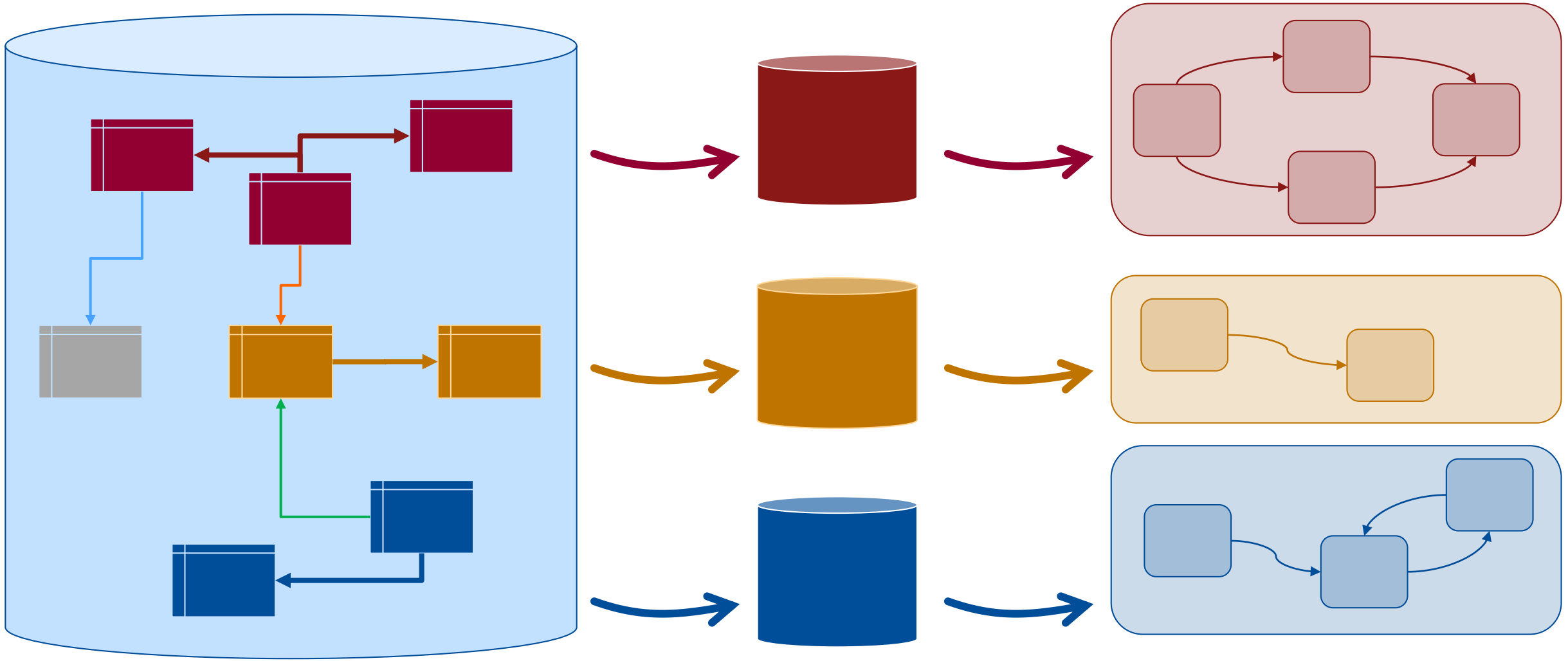


edges: 29 correct, 48 wrong
2 months SAP data

E. Nooijen, B. v. Dongen, and D. Fahland, “**Automatic Discovery of Data-Centric and Artifact-Centric Processes**,” in BPM Workshops. Springer, 2013, pp. 316–327.
Viara Popova, Dirk Fahland, Marlon Dumas: “**Artifact Lifecycle Discovery.”** Int. J. Cooperative Inf. Syst. 24(1) (2015)
Xixi Lu, Marijn Nagelkerke, Dirk Fahland: “**Discovering Interacting Artifacts from ERP systems**” IEEE Trans. On Services Computing DOI: 10.1109/TSC.2015.2474358

Discover Objects & Life-Cycles

[Nooijen et al. 2013, Lu et al. 2015]



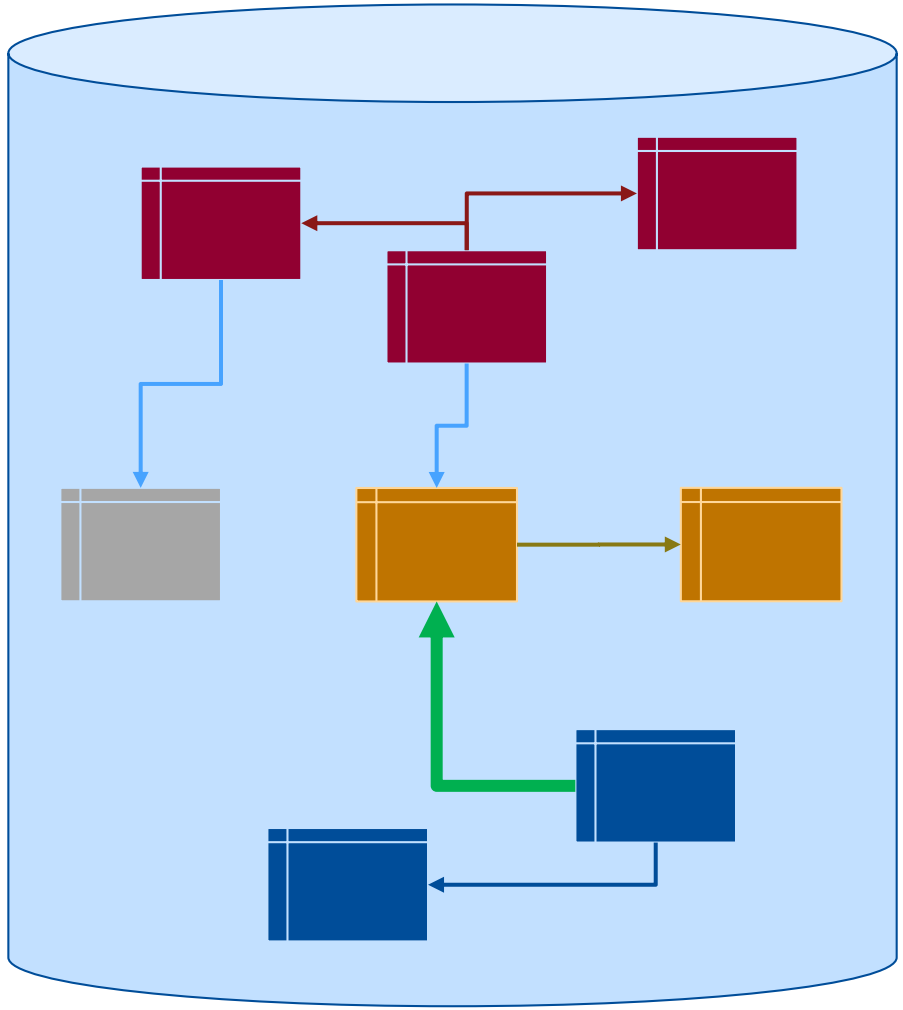
cluster tables

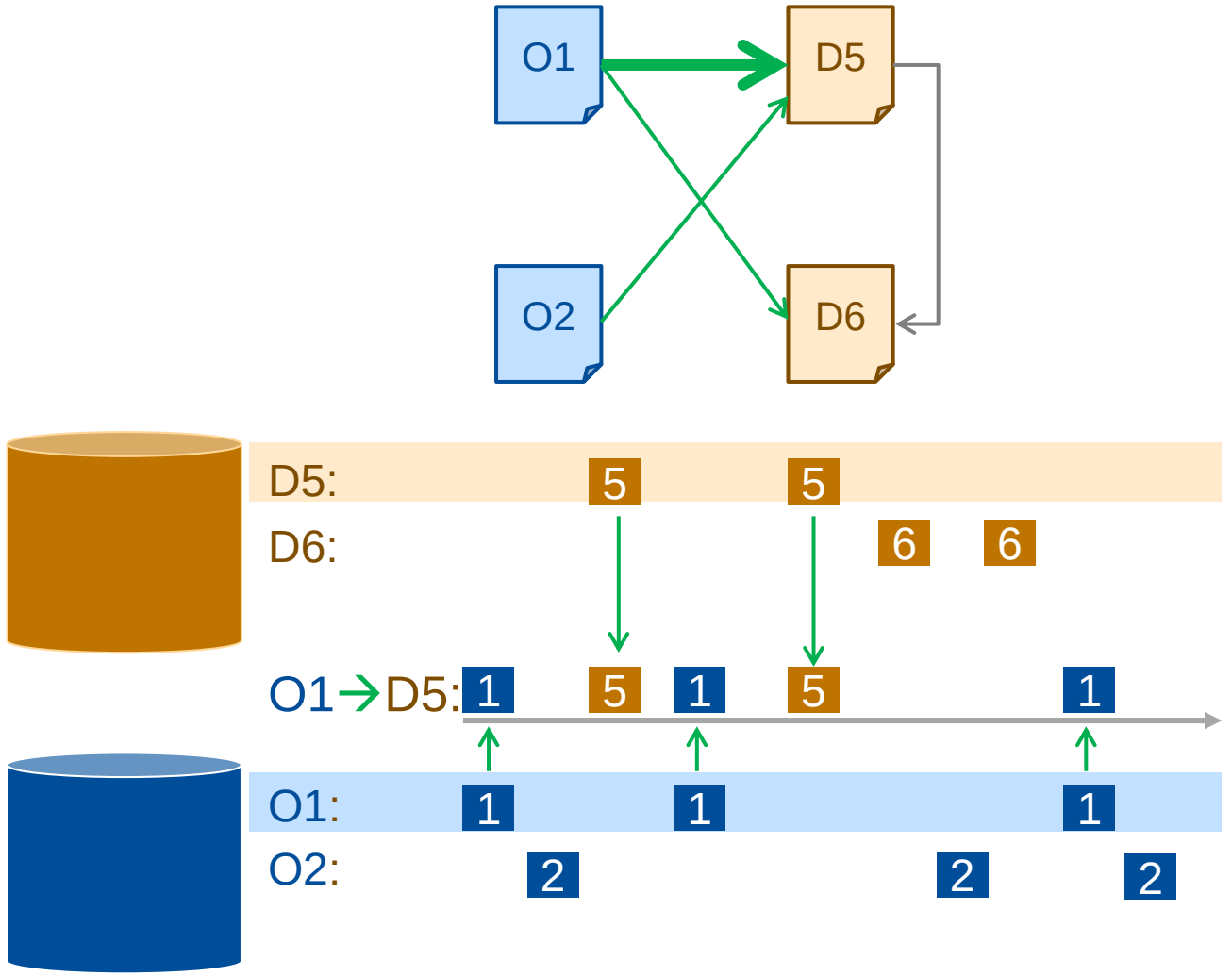
extract event logs

discover life-cycle model
(classical discovery)

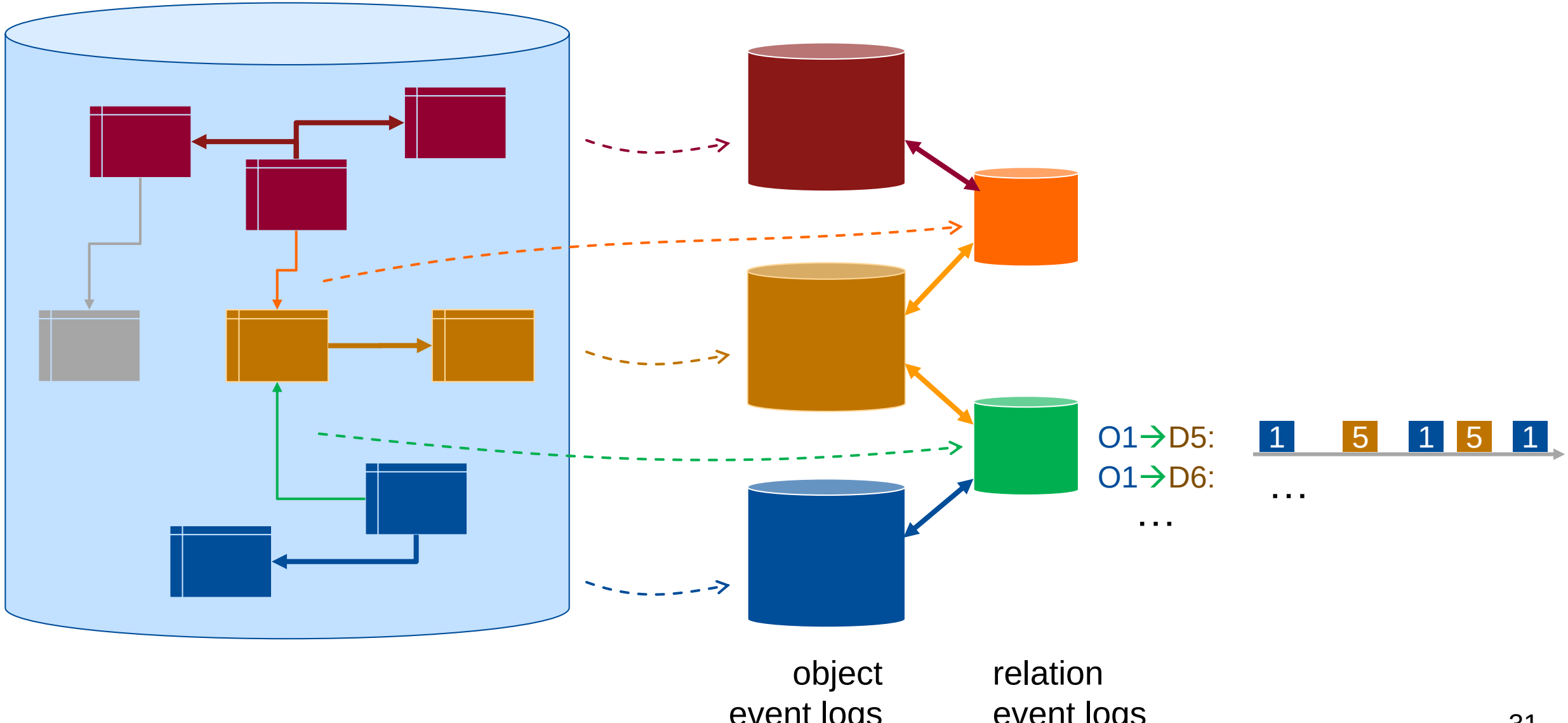
… and Relations

[Lu et al. 2015]

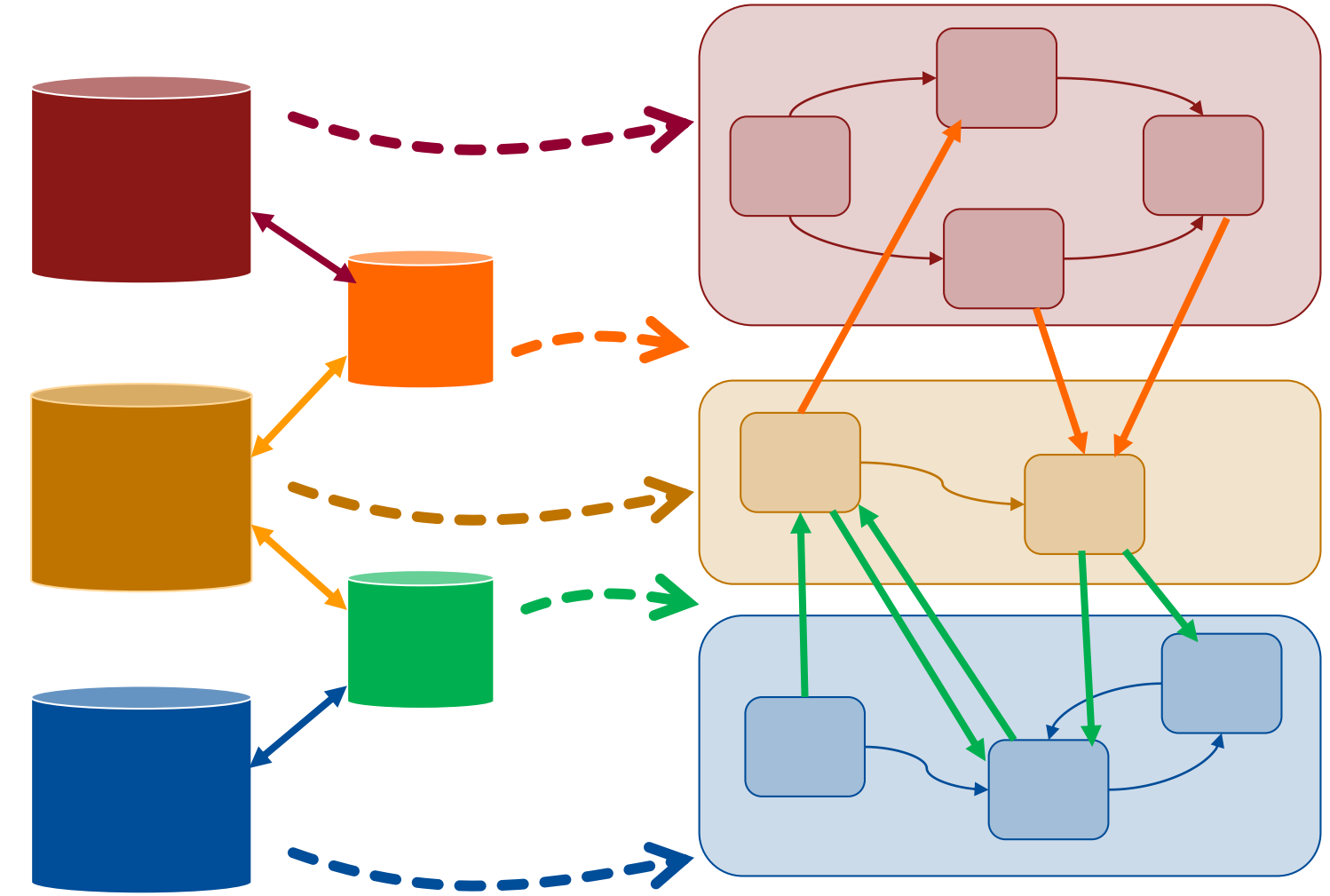




Network of Event logs (linked via shared events)



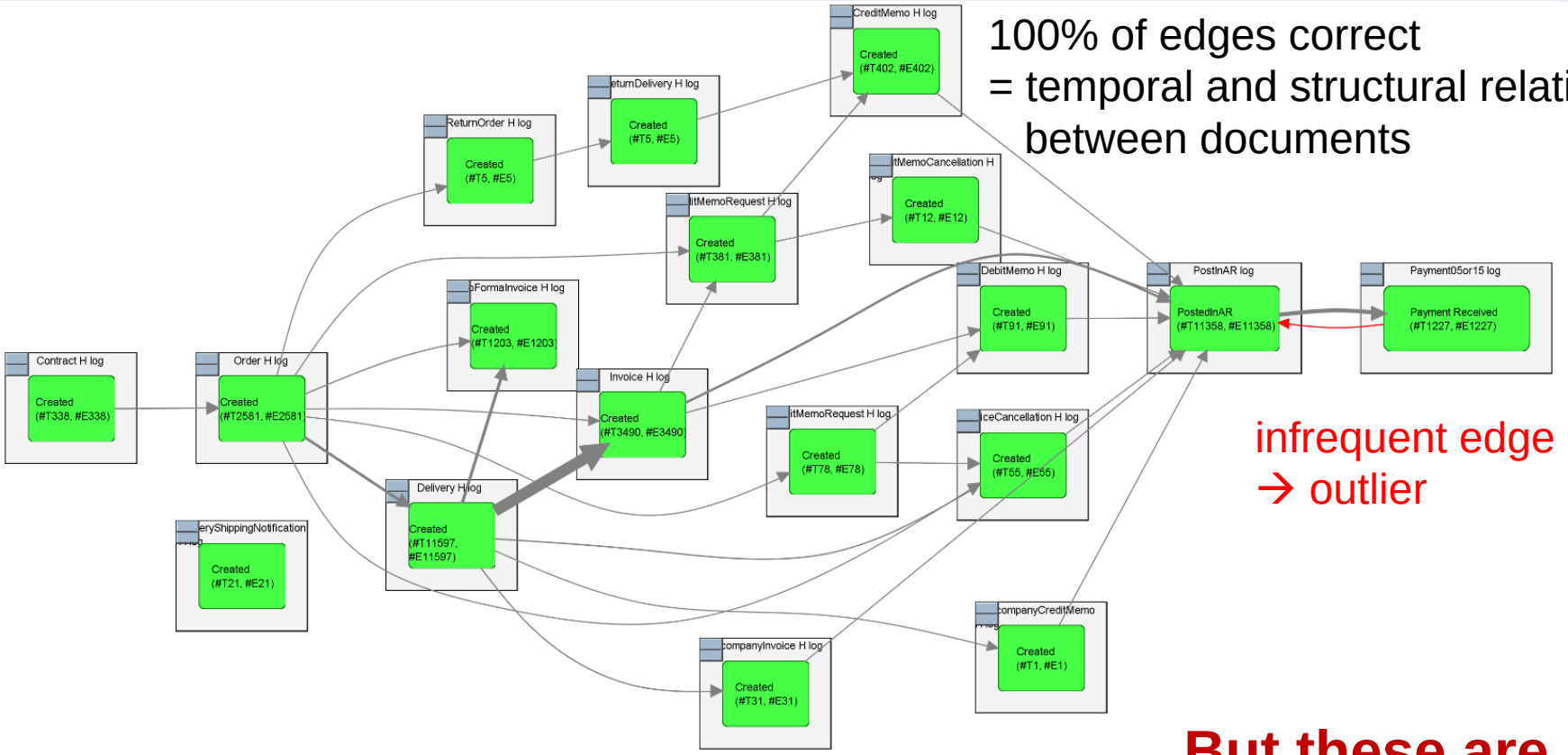
Process Mining in Networks of Event Logs



object event logs

relation event logs

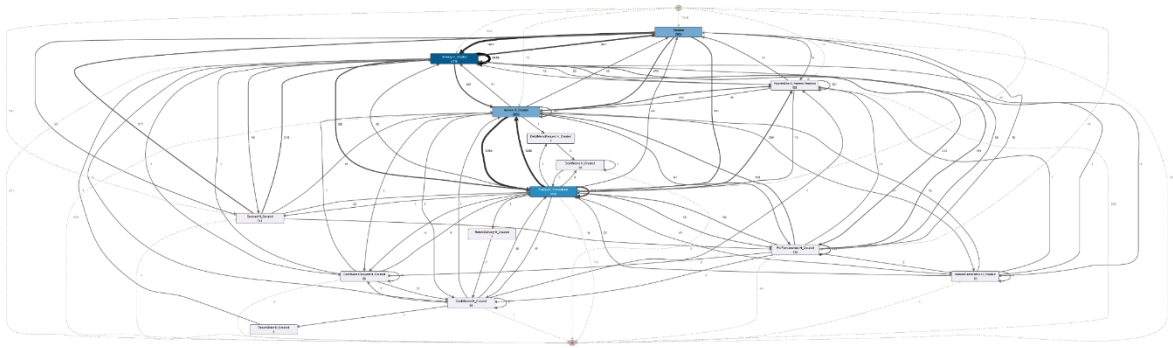
SAP Order-To-Cash "Artifact-Centric Model"



100% of edges correct
= temporal and structural relation between documents

infrequent edge
→ outlier

But these are just pictures!



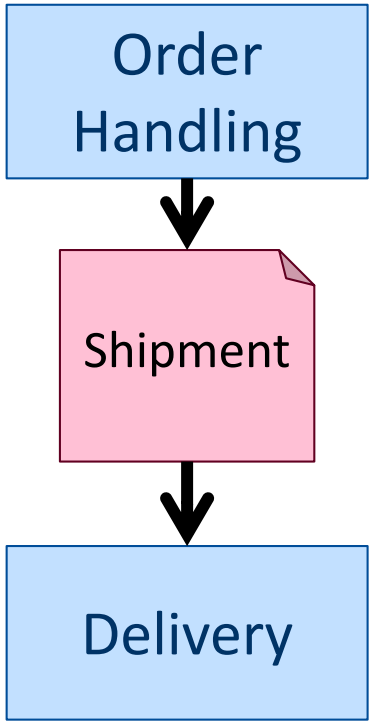
29/77 correct edges

Understandable Multi-Dimensional Processes

high-level process model

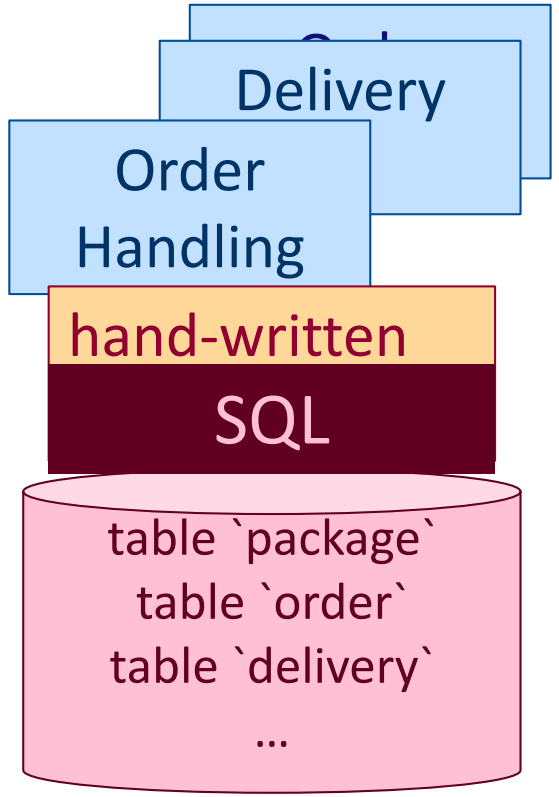
same?

implemented processes



simple to understand but too imprecise

precise but non-understandable, non-analyzable

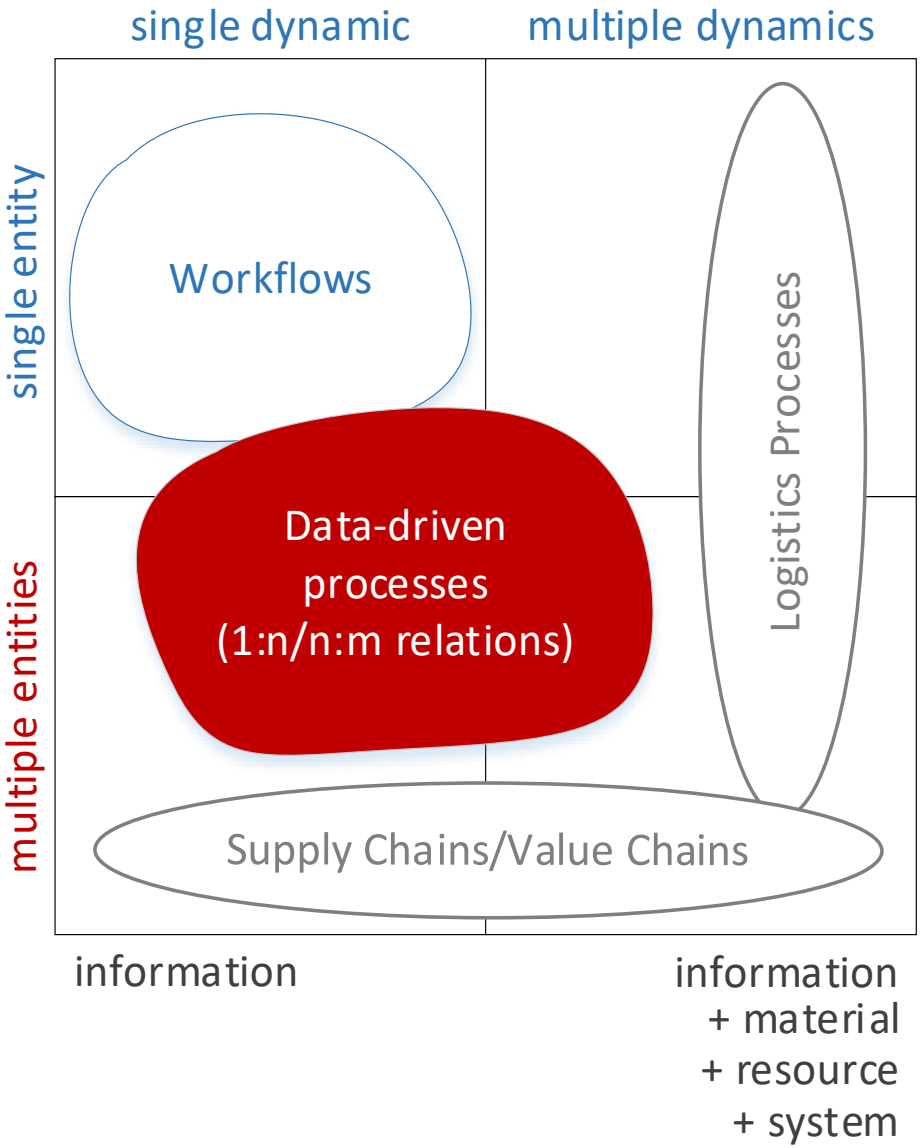


complexity

abstract graphical concepts

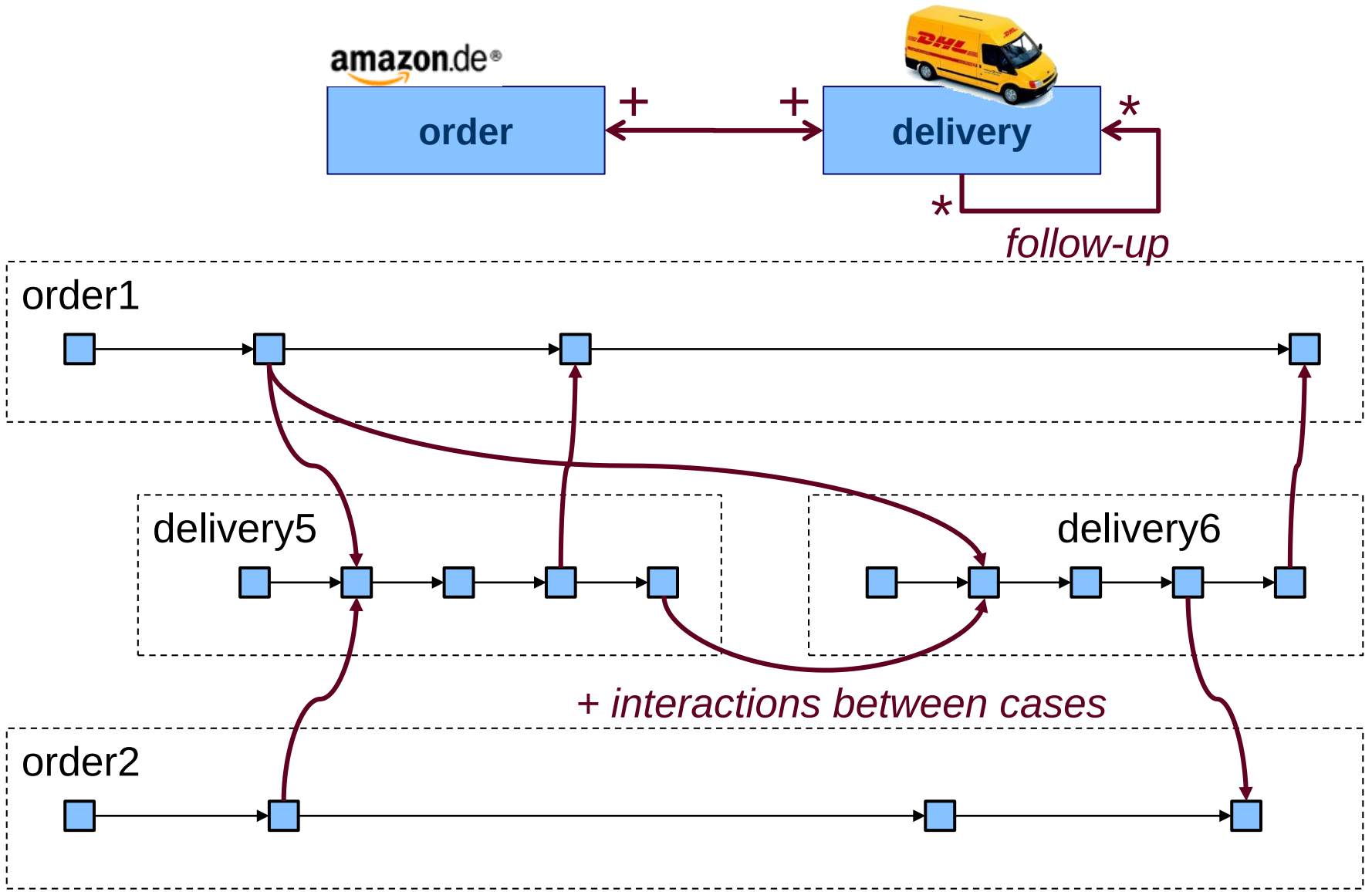
written code

Multi-Dimensional Processes

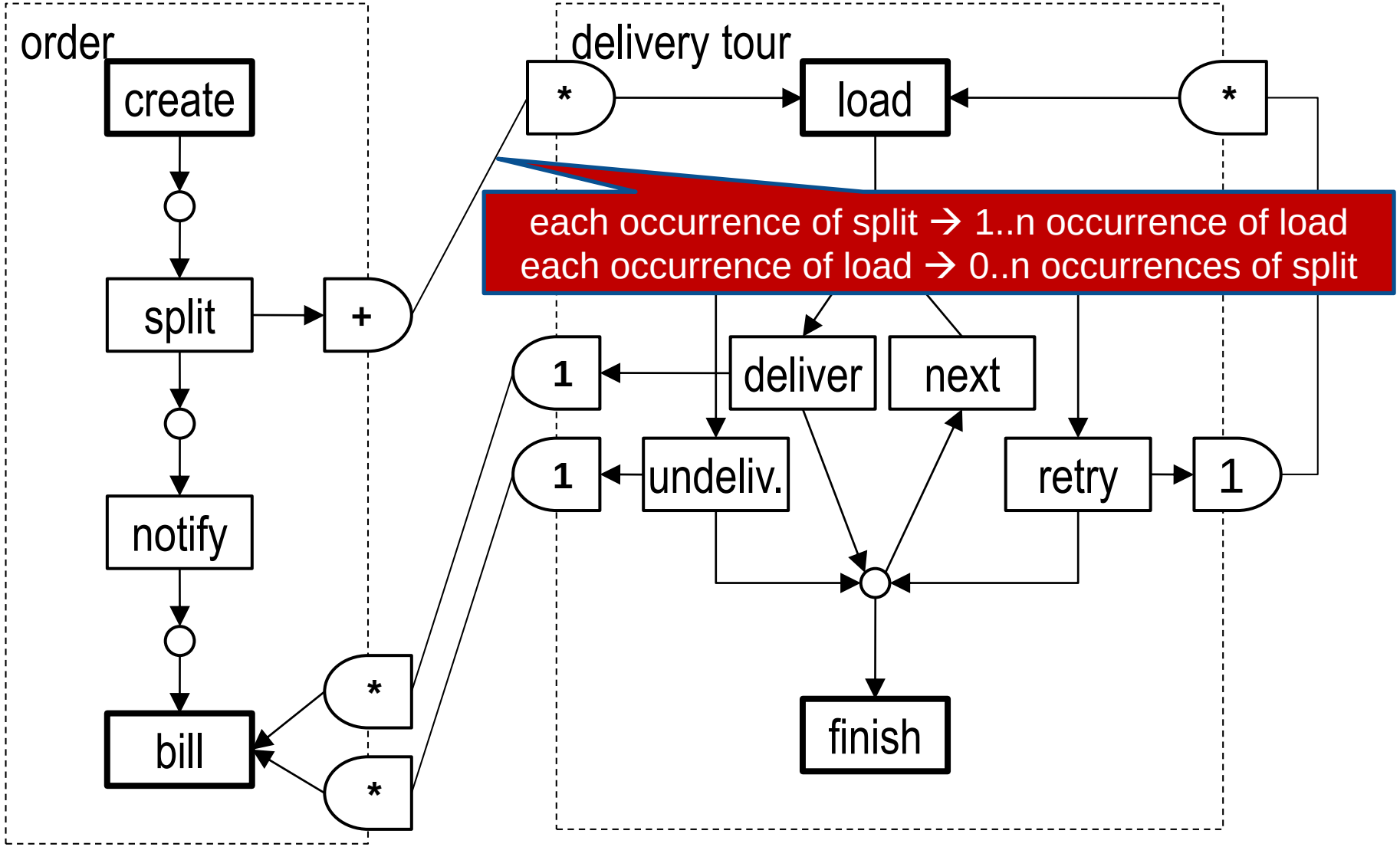


- Processes with multiple entities
  - How do they look like in practice?
  - Where does current modeling fail?
    - Implementation
    - Process mining
  - **A proposal**
  - New research questions

Need a model that can produce such a run:

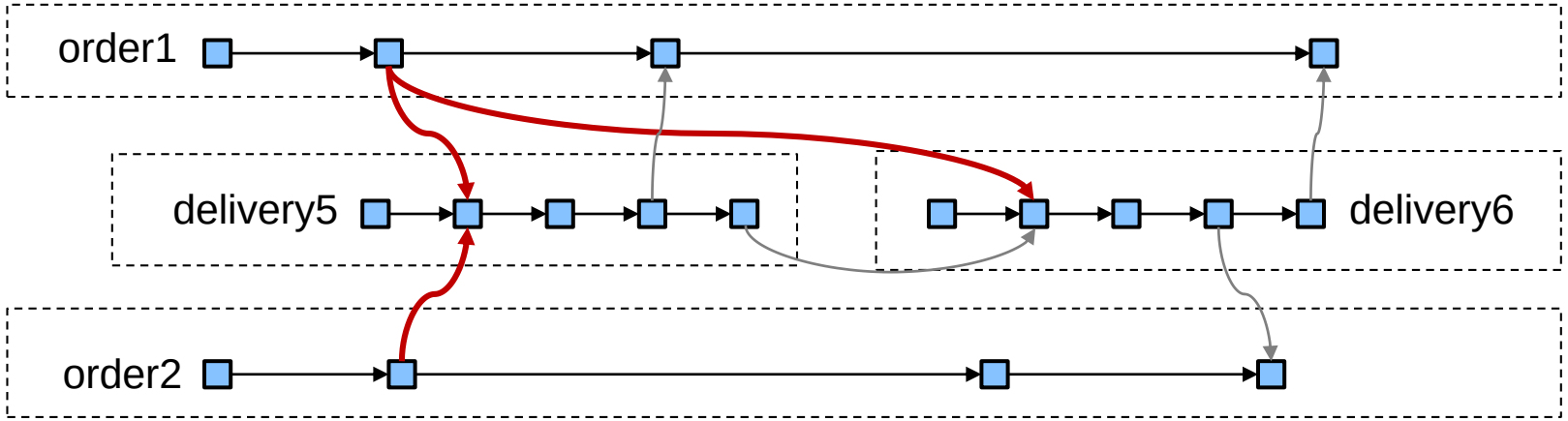


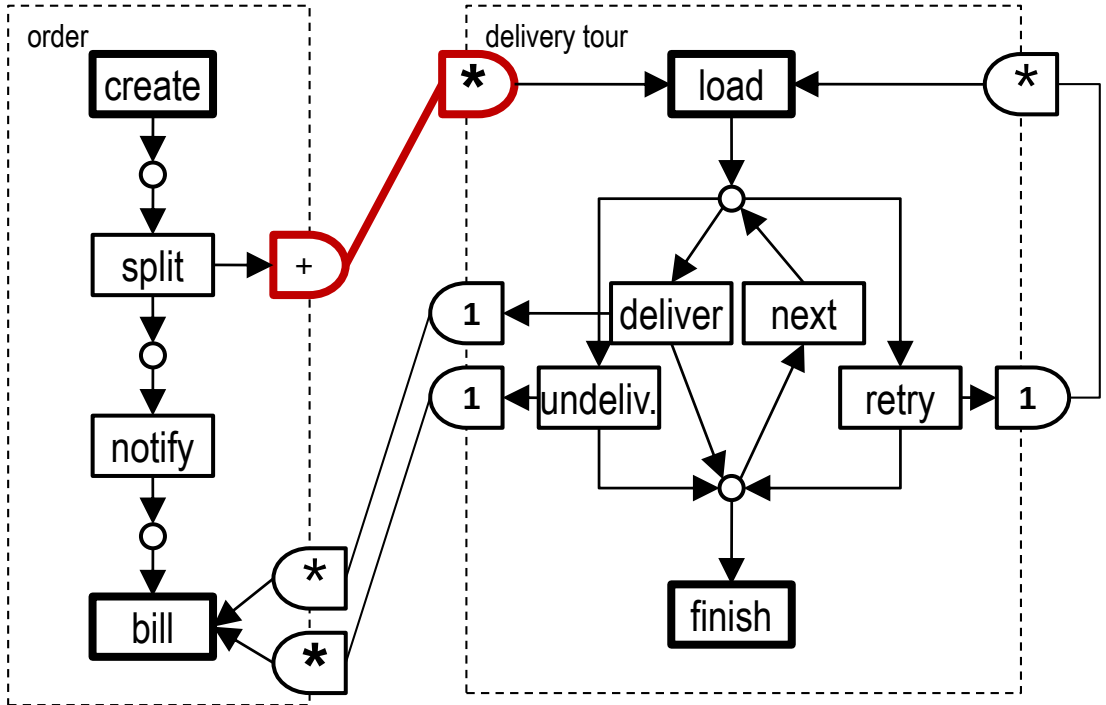
Proclets: Cardinalities between events



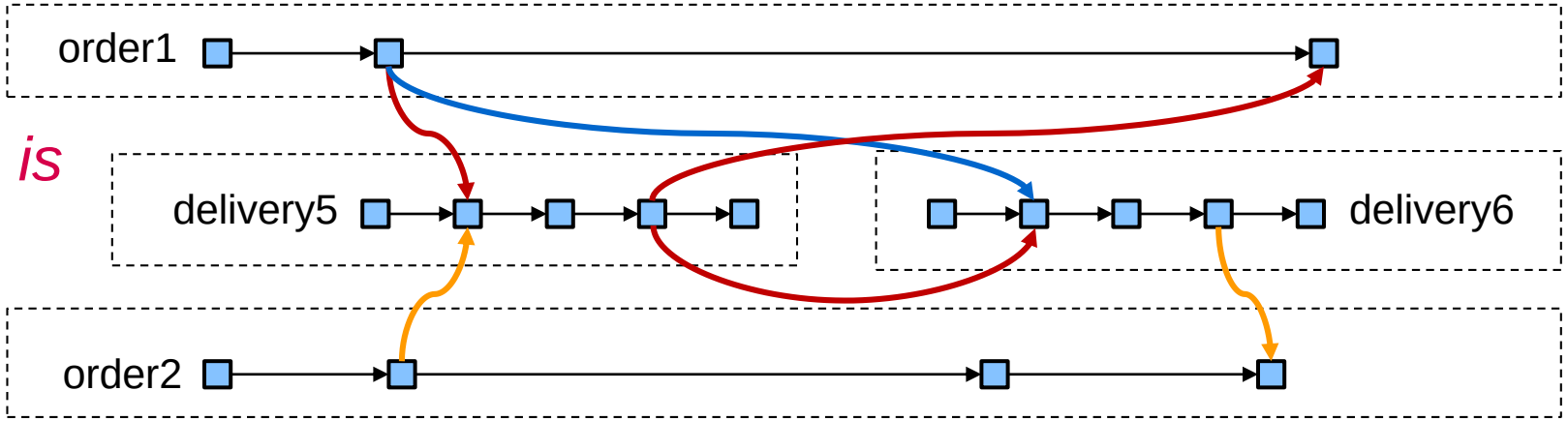
[van der Aalst 2001]

Cardinalities constrain possible n:m interactions





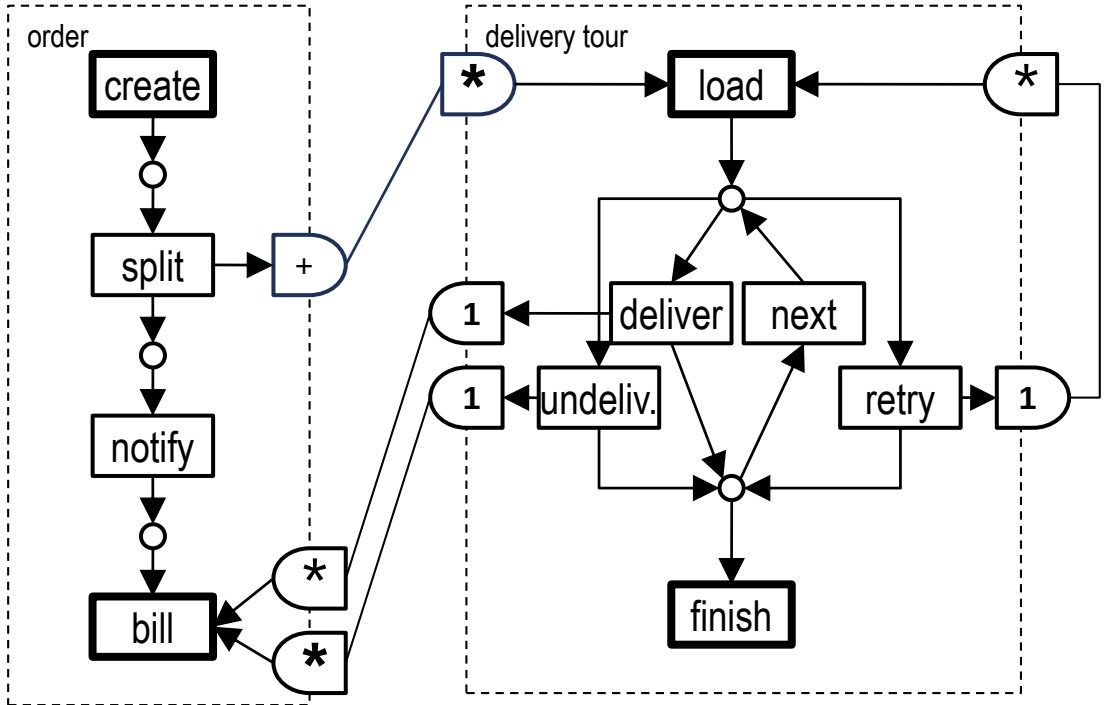
Problem: Cardinalities do NOT preserve objects



package21 is duplicated

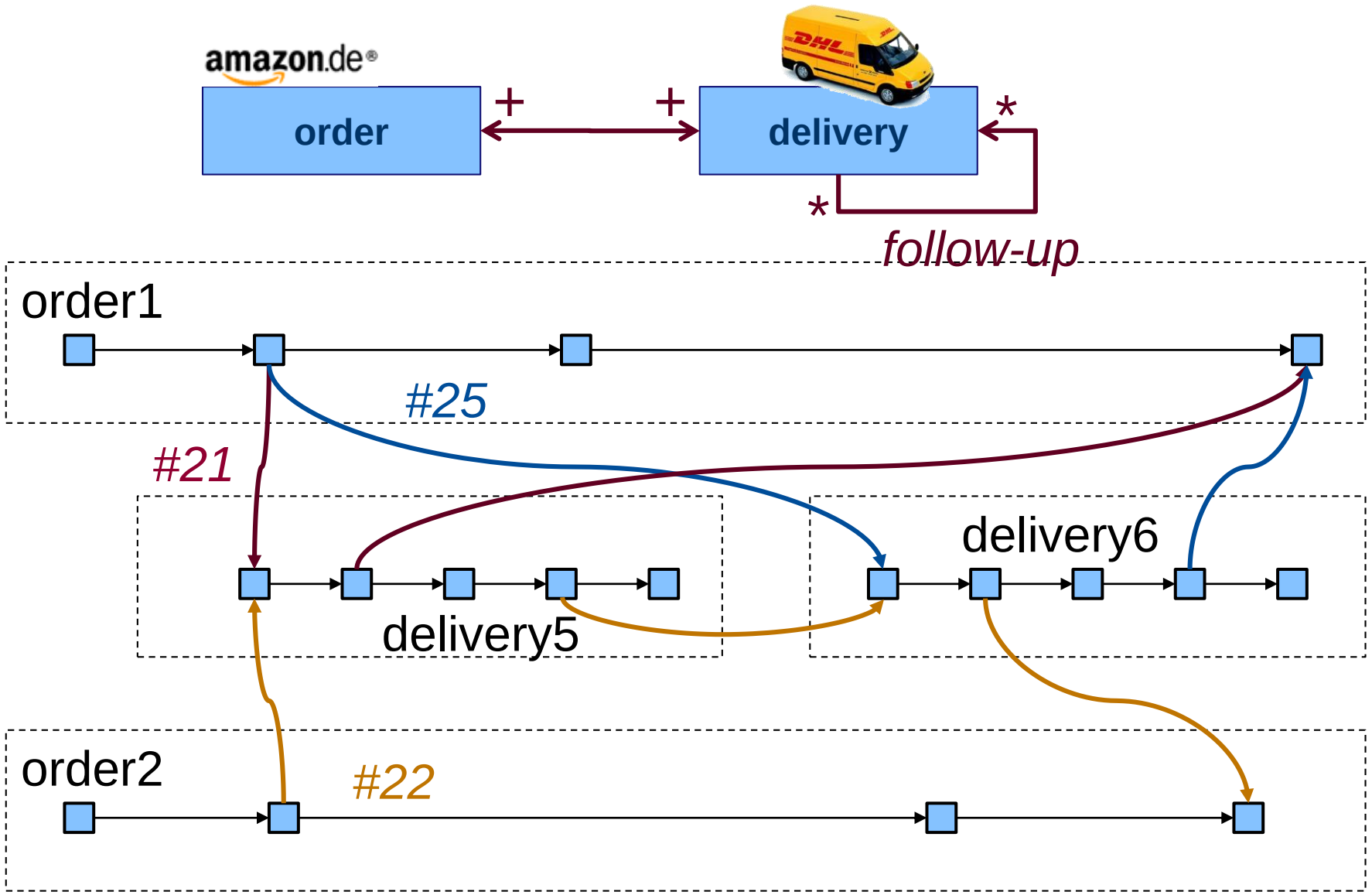
package25 is lost

package22 teleports

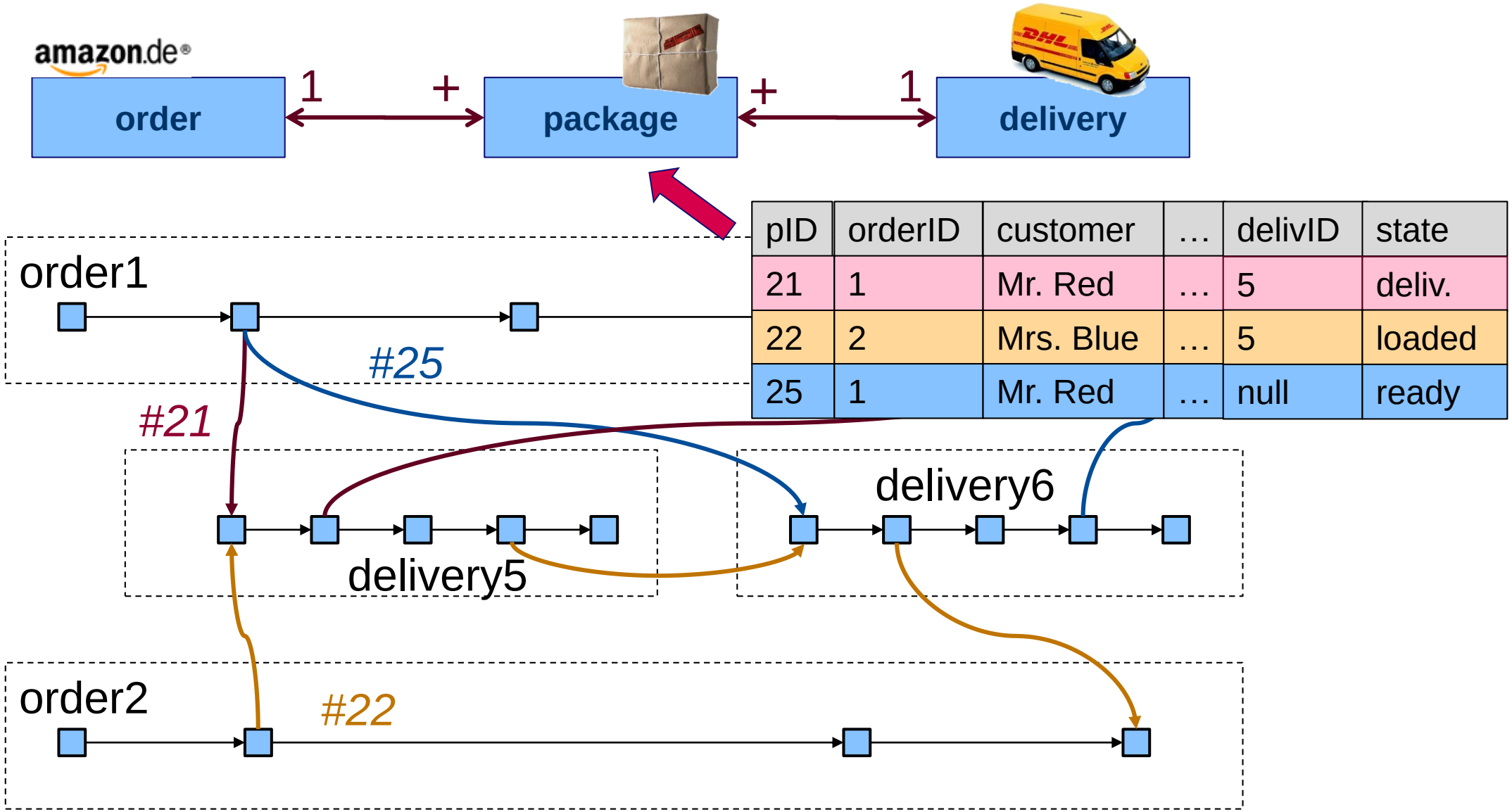


… but cardinality constraints satisfied

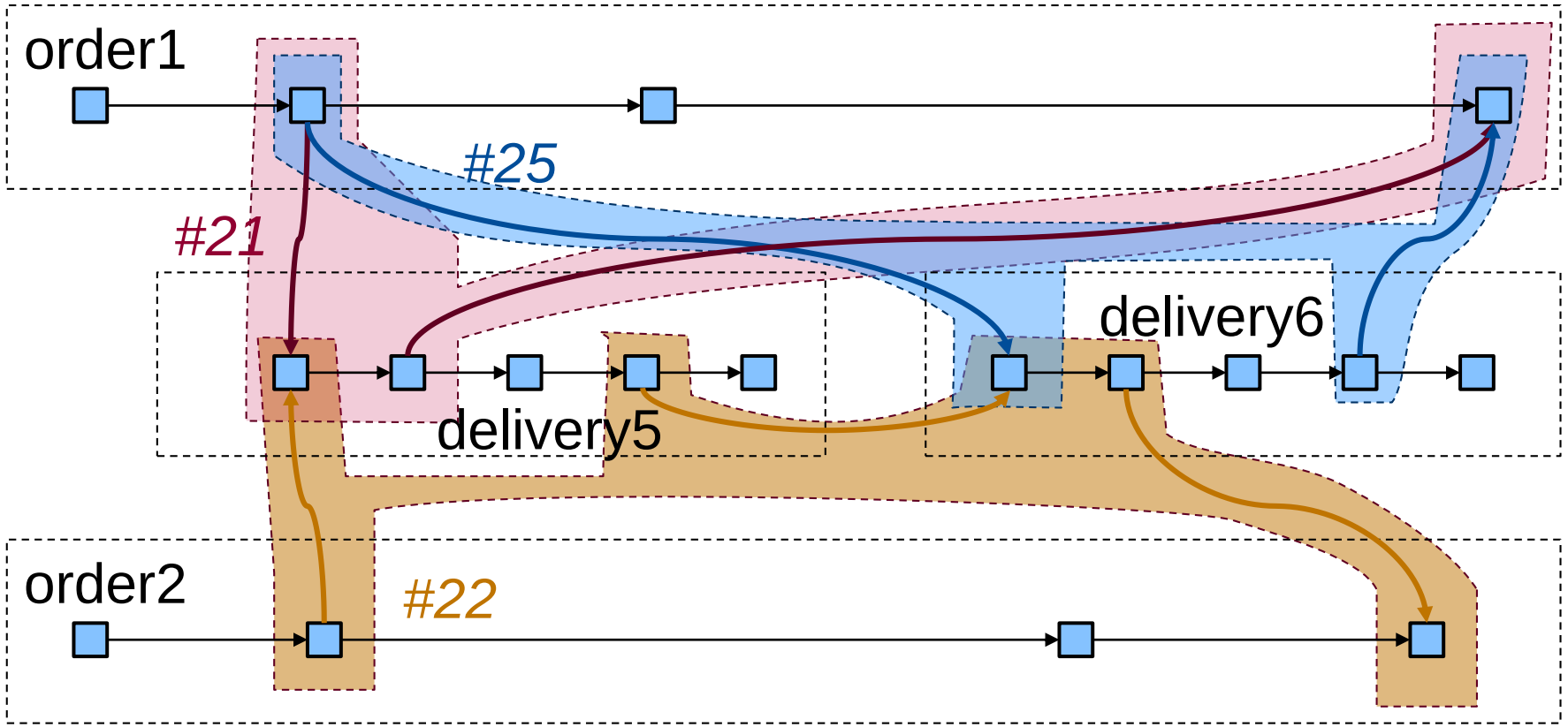
So what's the problem with M:N relationships?



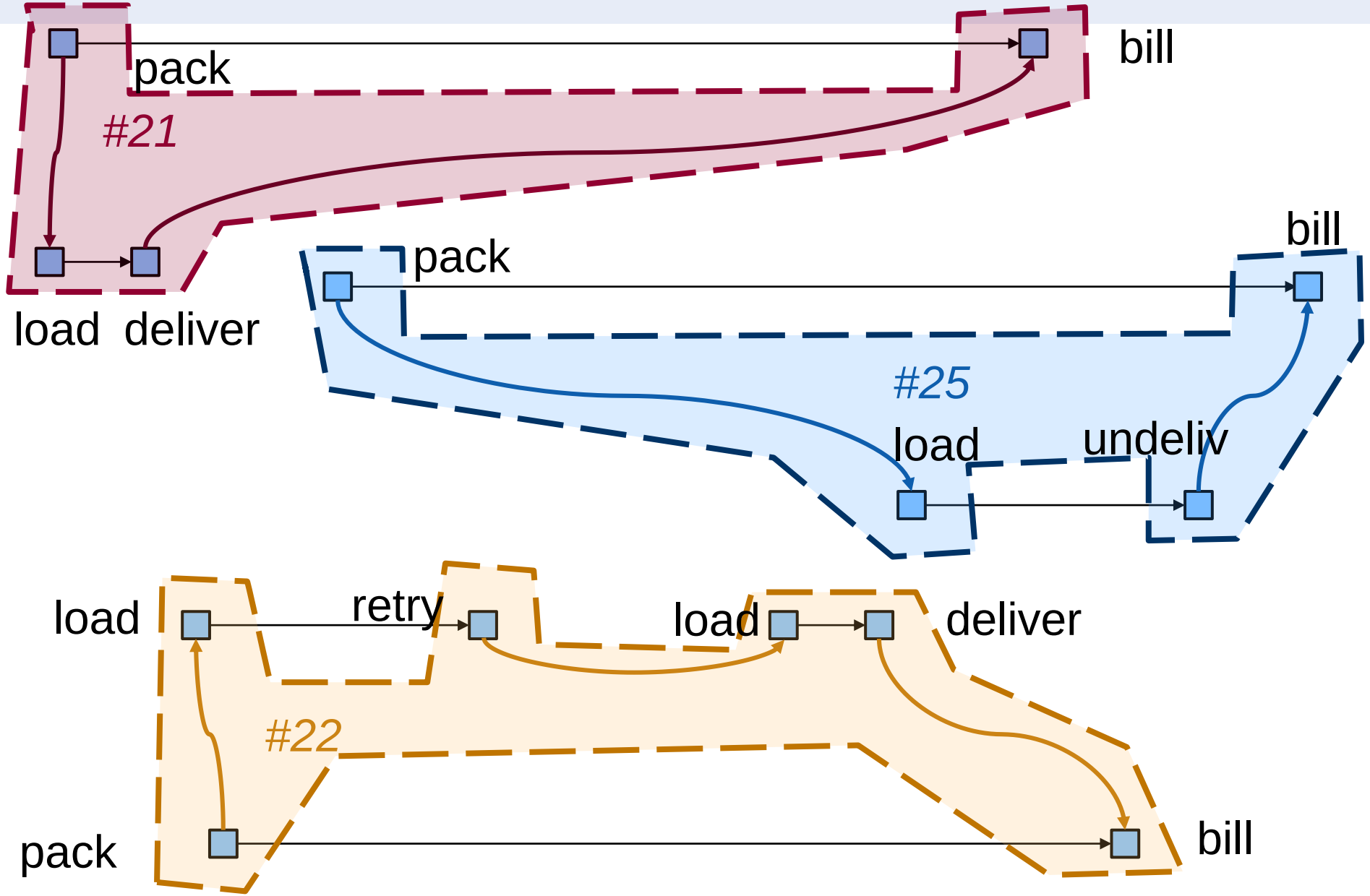
Reify Relation → 2nd Normal Form



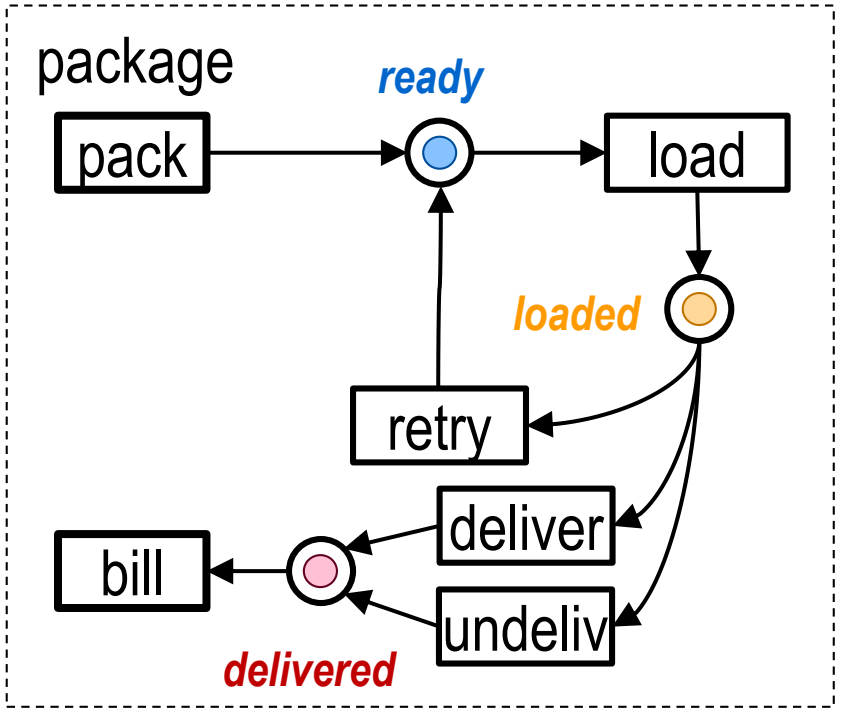
Reify Relation → Reify Behavior



3 Conversations = Behavior of an Object “Package”

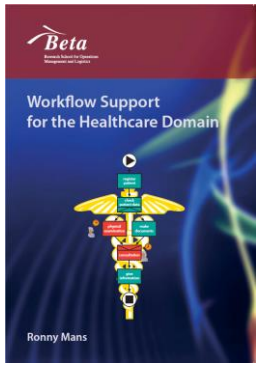


Behavioral Model for Object Package



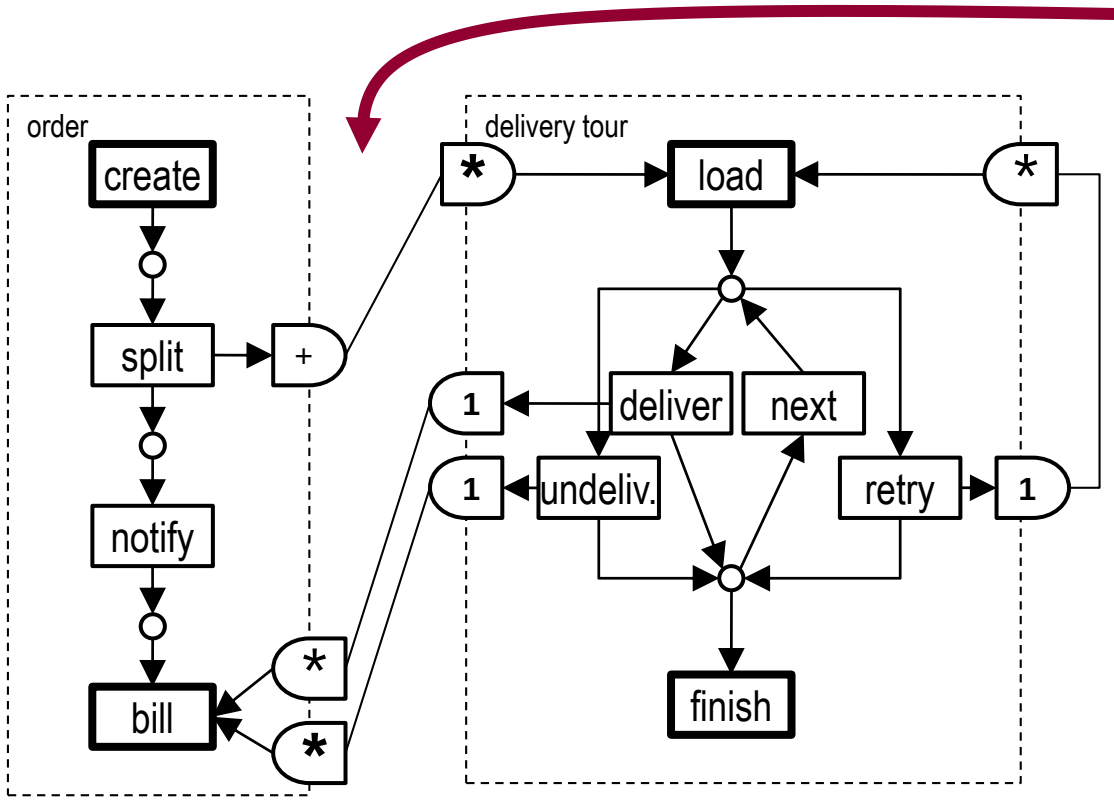
states in life-cycle model abstract from attribute values in data model

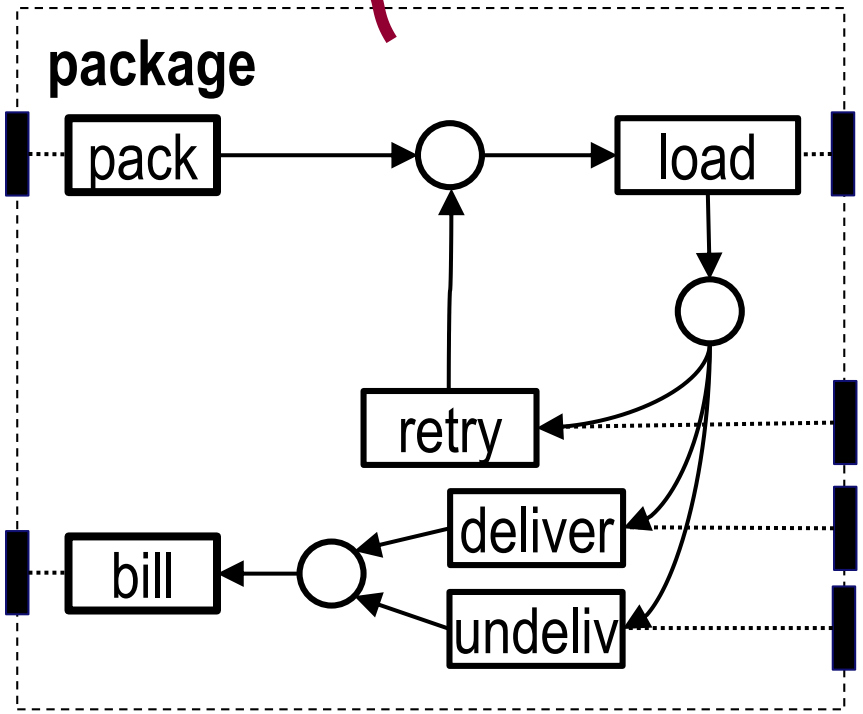
pID	orderID	customer	…	delivID	**state**
21	1	Mr. Red	…	5	**deliv.**
22	2	Mrs. Blue	…	5	**loaded**
25	1	Mr. Red	…	null	**ready**



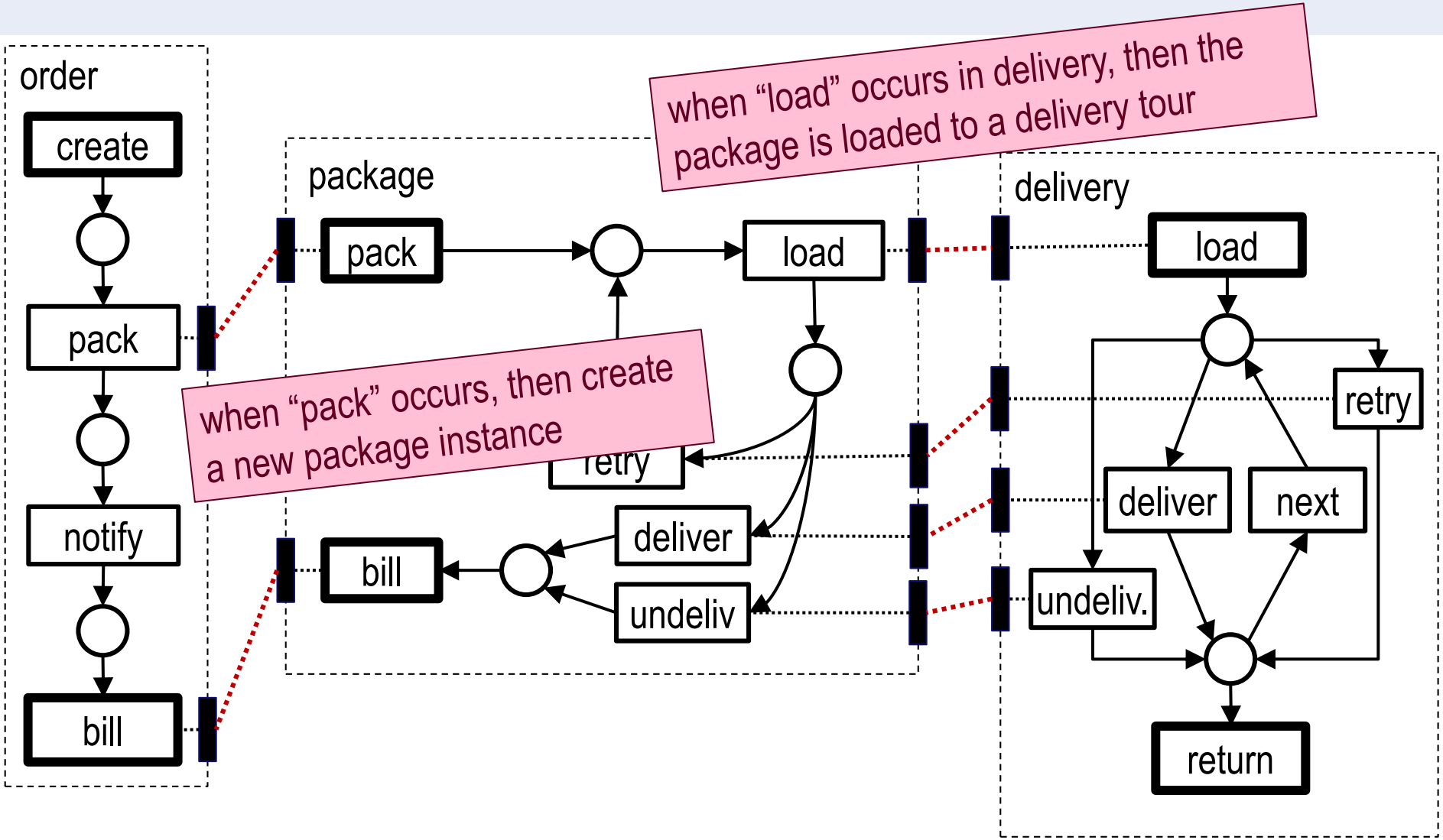
Precursor to this idea in: Ronny Mans. Workflow Support for the Healthcare Domain. PhD Thesis. 2011, Ch. 6.

Reify Behavioral Model → Normal Form

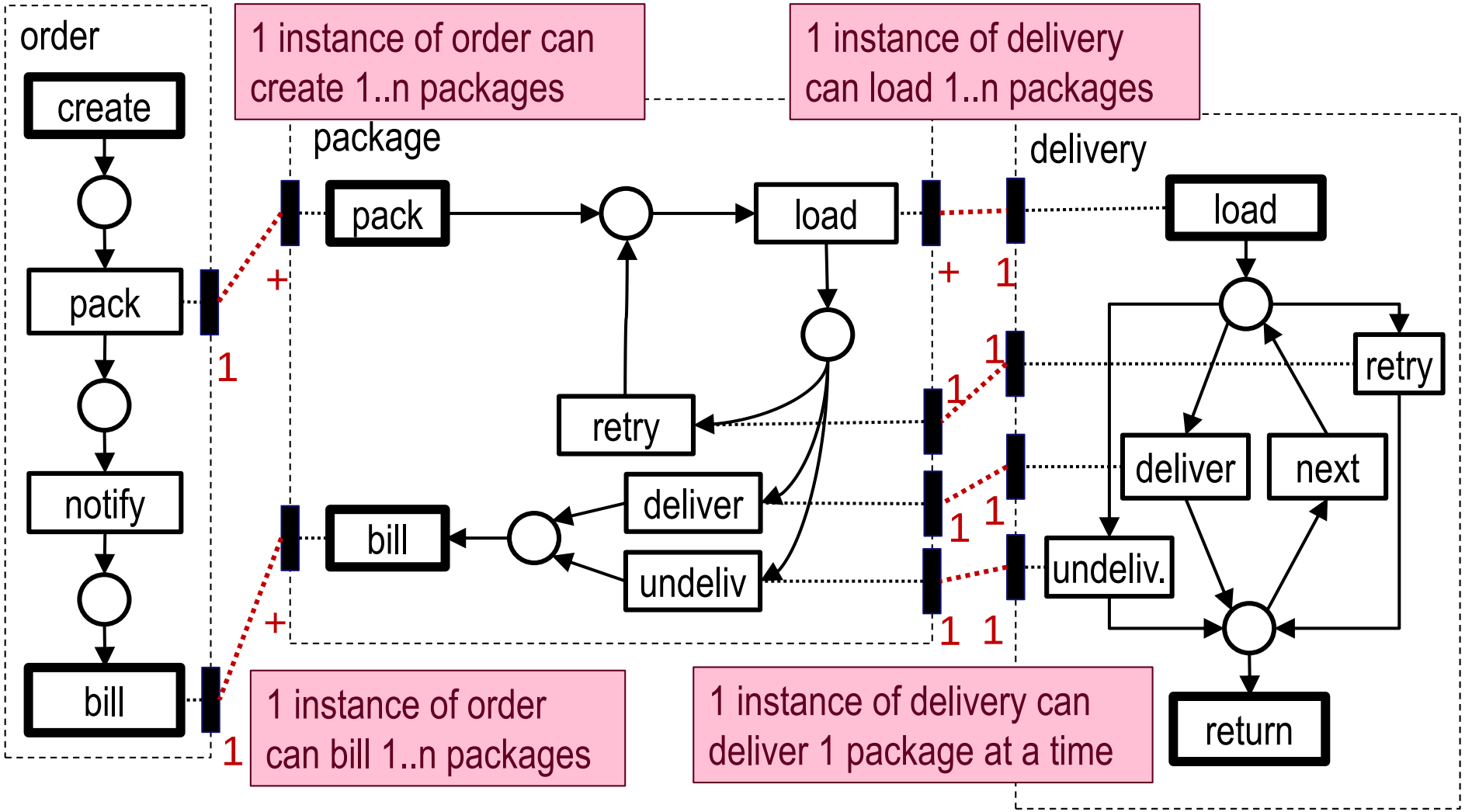




Behavioral Normal Form: Synchronous Interaction



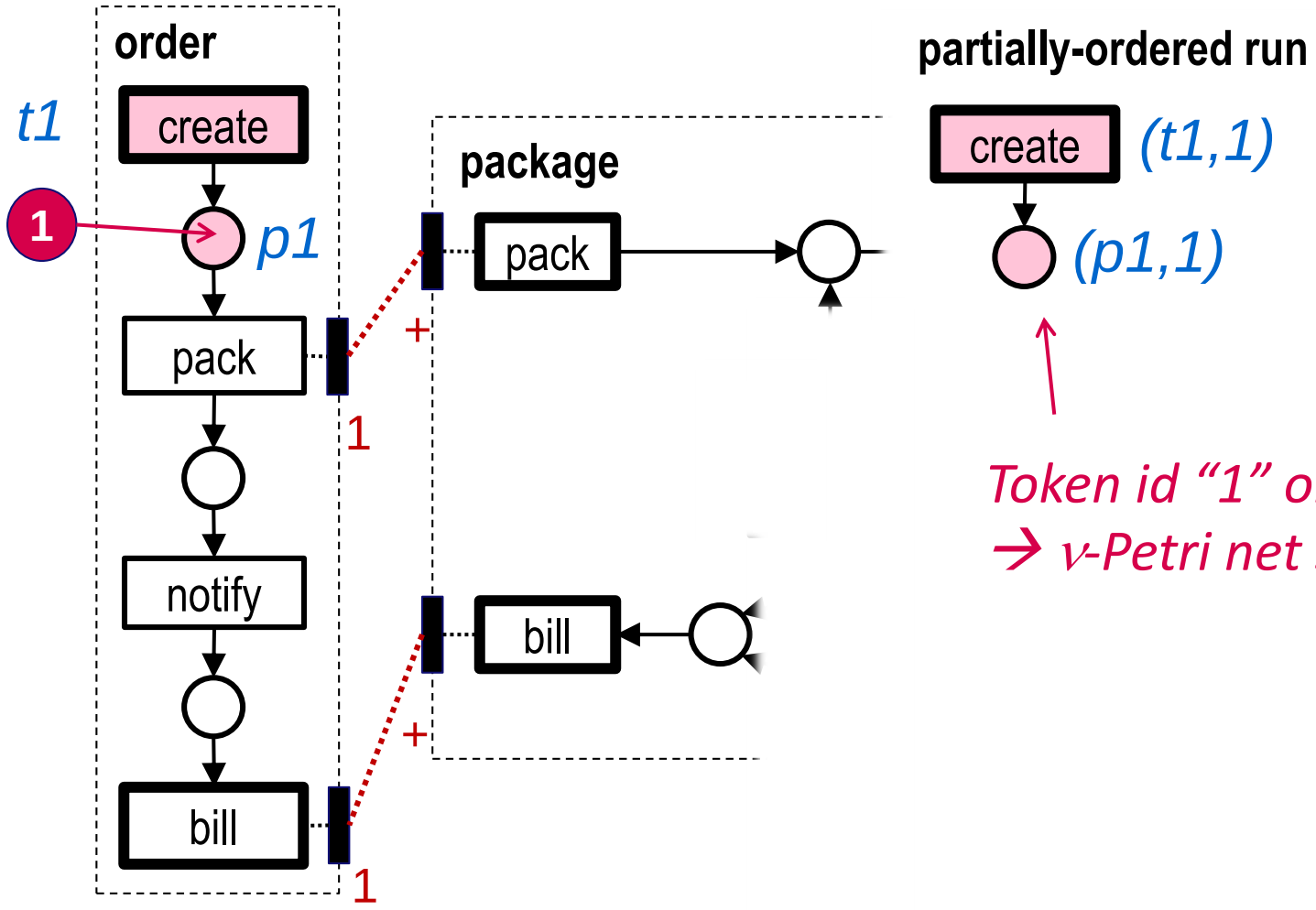
… with 1:1/1:N synchronization only: Normal Form!



formal model in paper is general, allows N:M synchronization

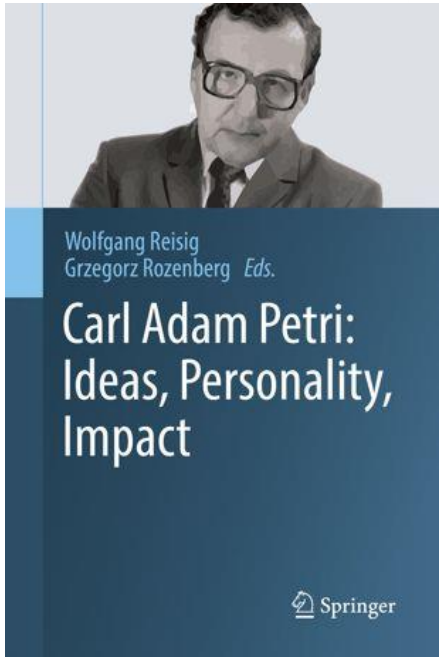
Dirk Fahland "Describing Behavior of Processes with Many-to-Many Interactions", Petri Nets 2019

Semantics: As “net-affine” as I could make it



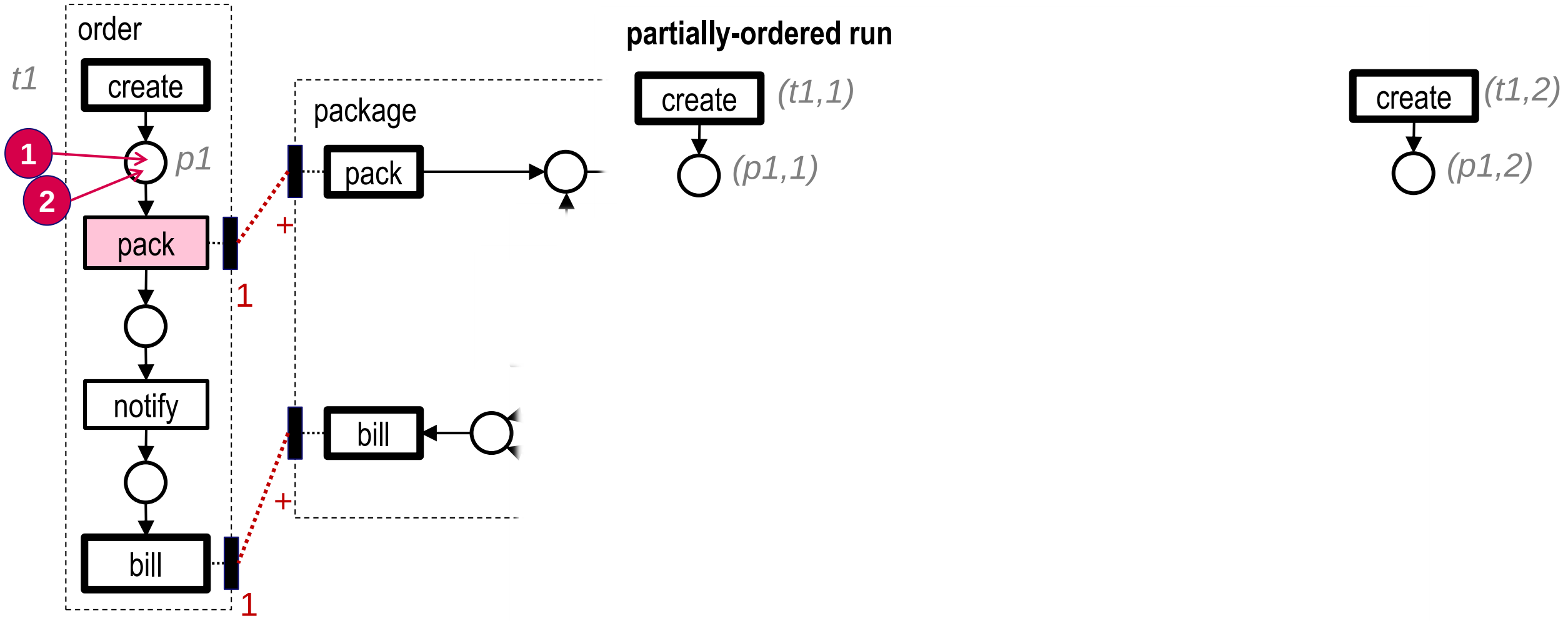
*In instance “1” of process **order** transition **create**”occurred*

Token id “1” on place p1
→ ν-Petri net semantics

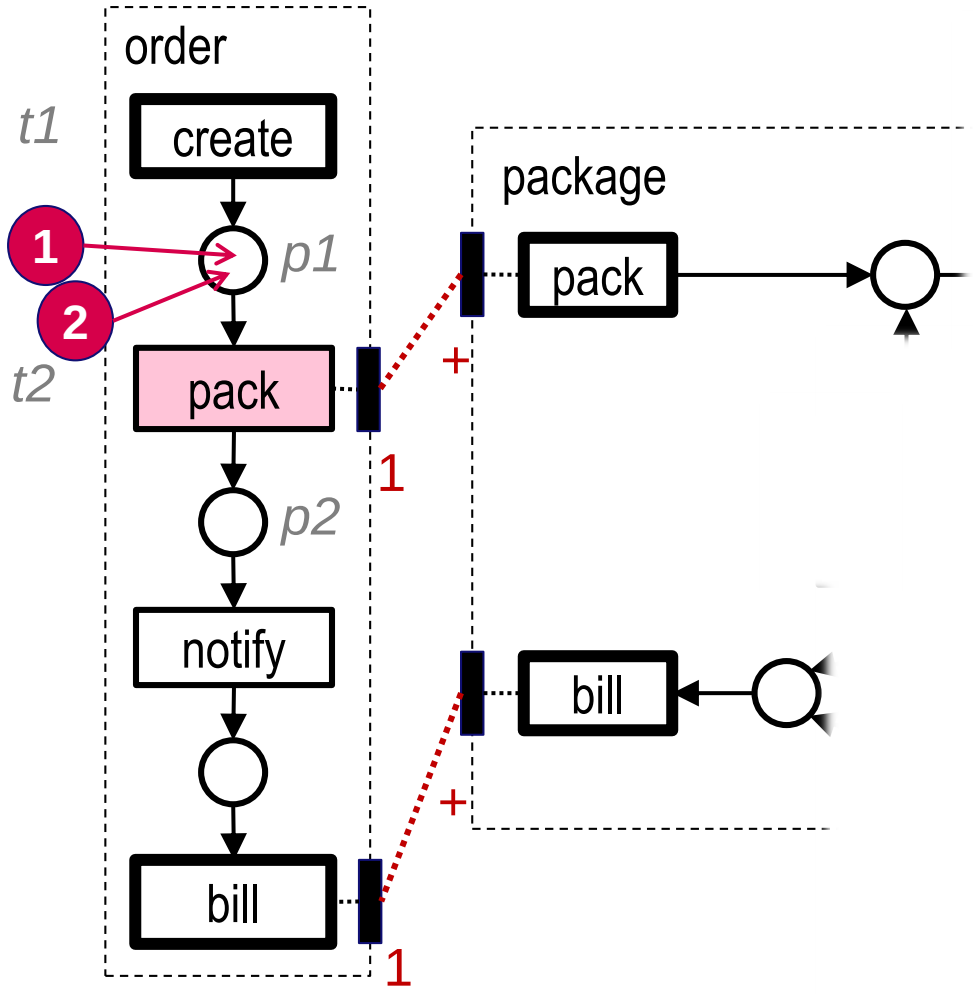


Dirk Fahland: Petri's Understanding of Nets.
Carl Adam Petri: Ideas, Personality, Impact 2019: 31-36

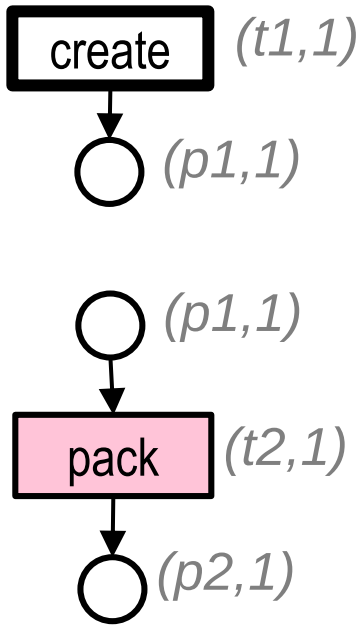
Different Instances: “own” local runs

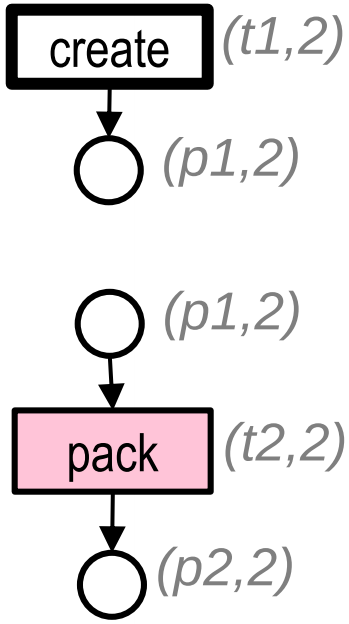


Different Instances: “own” local runs



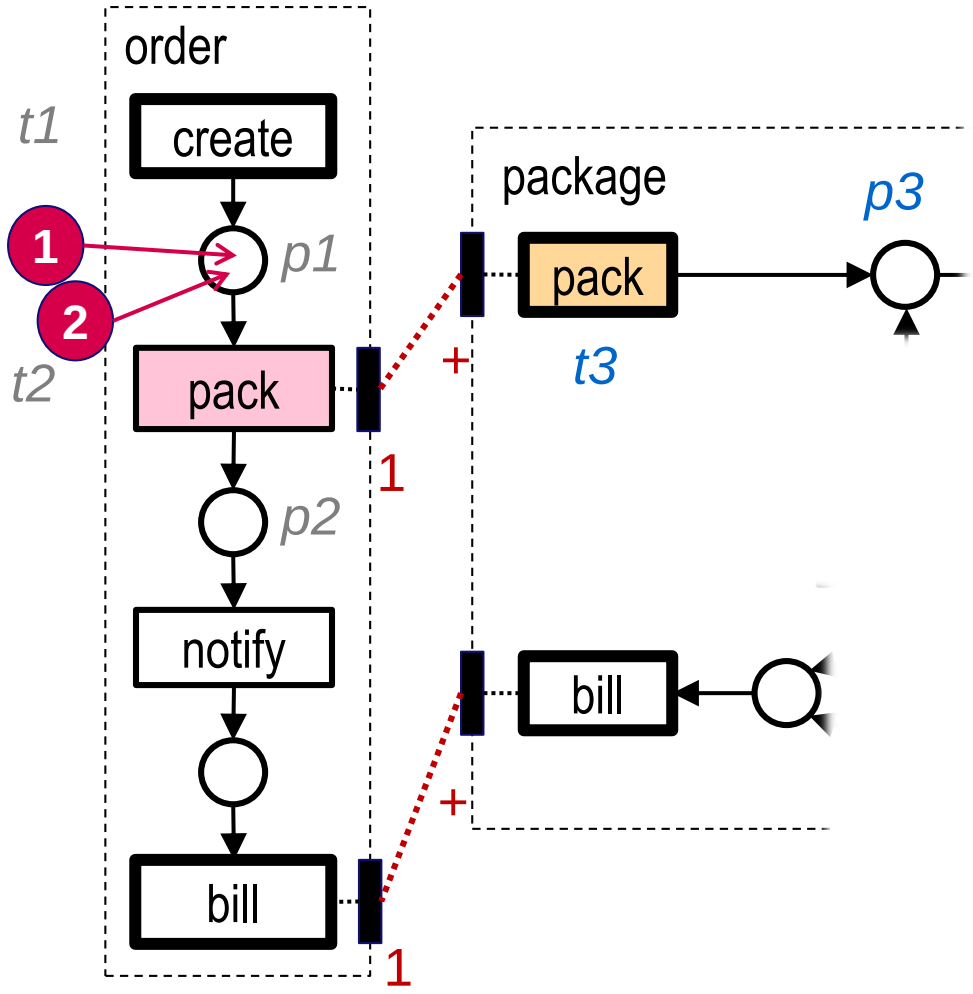
partially-ordered run



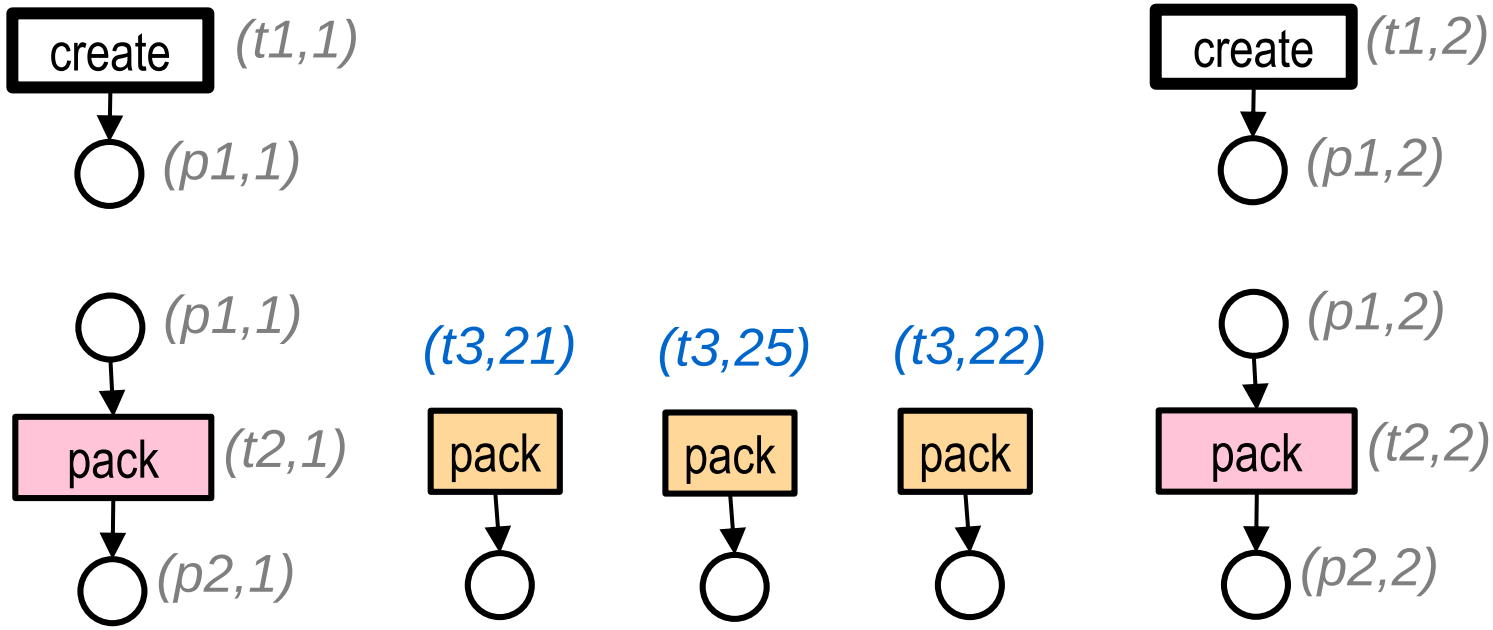


possible extensions of the run

How to synchronize instances?

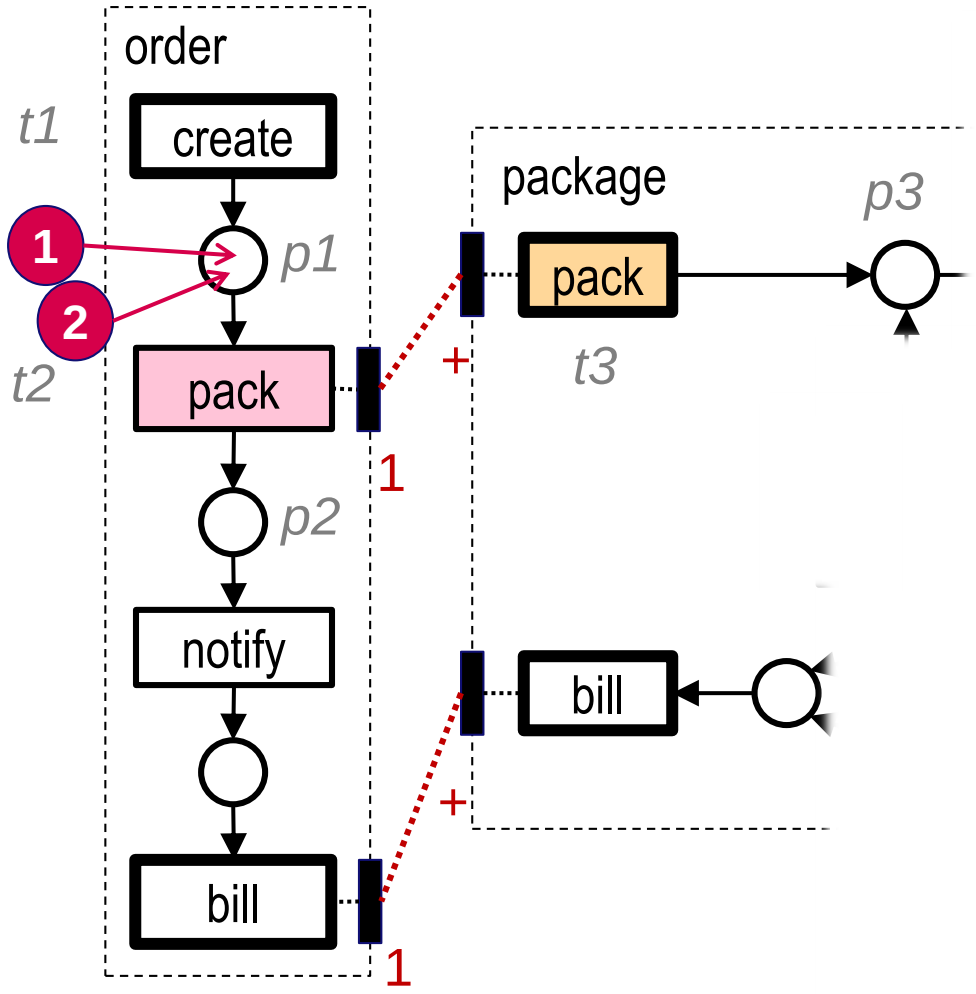


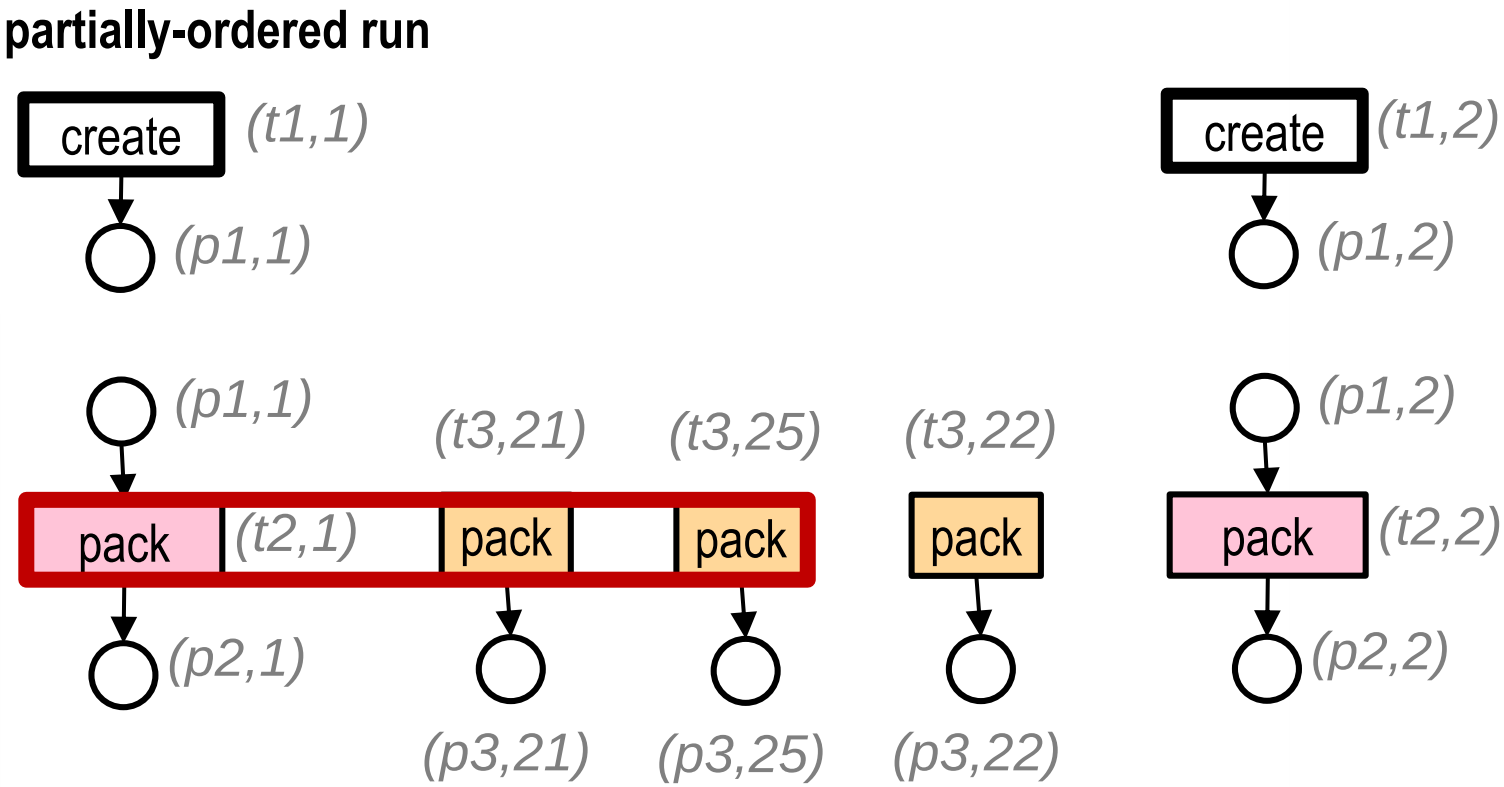
partially-ordered run



possible extensions of the run

Binding = non-det. chosen set of ids to sync

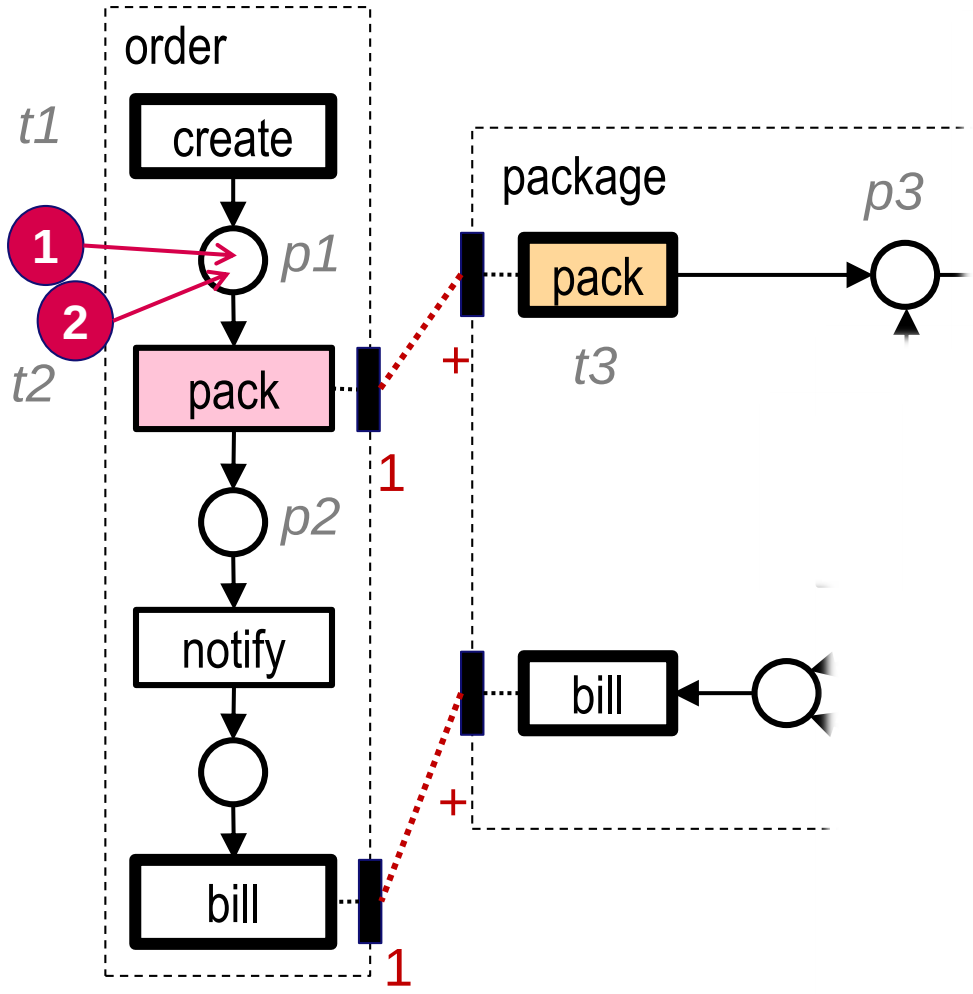


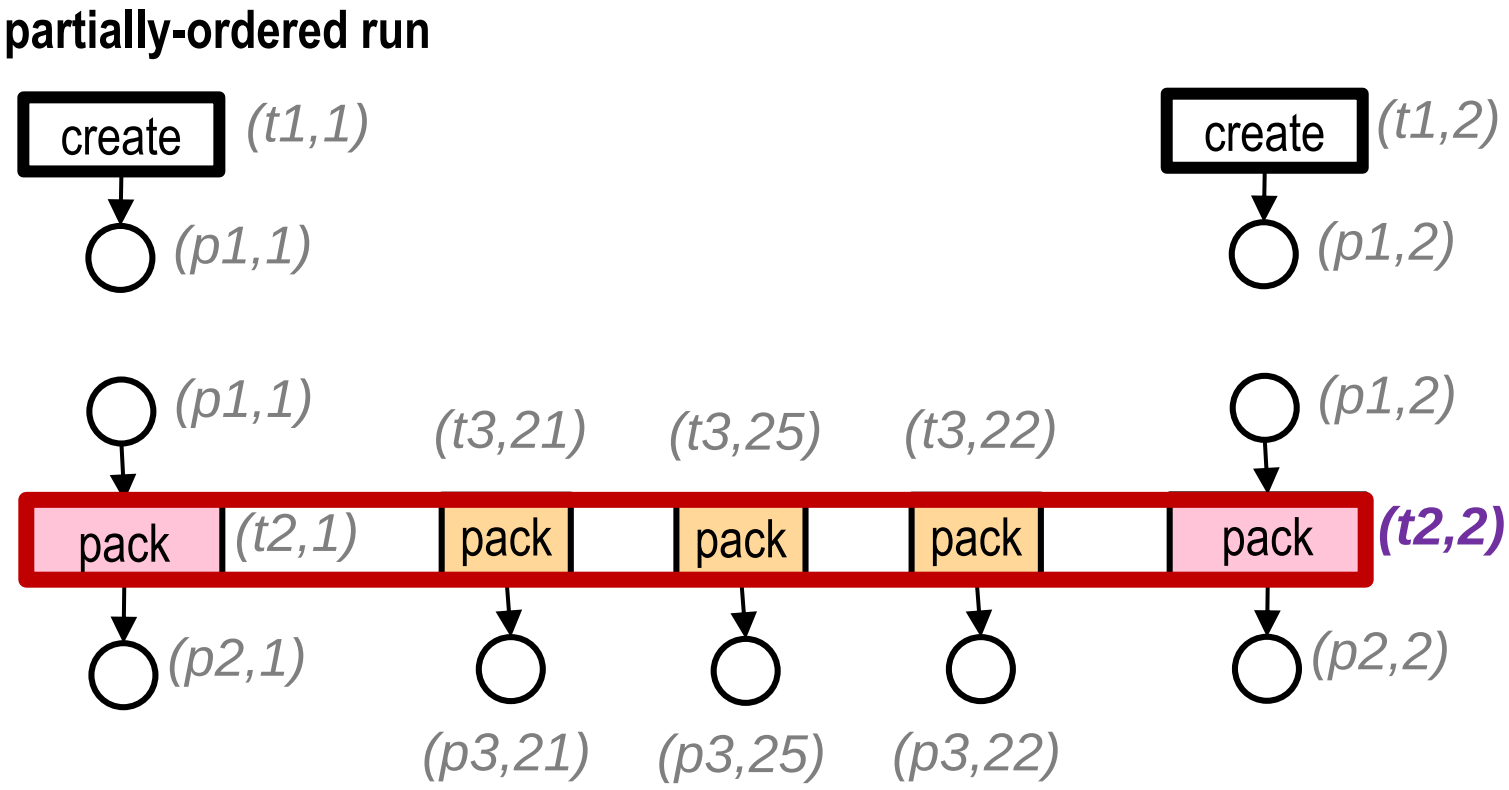


***binding* at channel (pack,pack):**
{ 1,21,25 }

***relation* underlying the binding**
{ (1,21),(1,25) }

Binding = non-det. chosen set of ids to sync





***binding* at channel (pack,pack):**

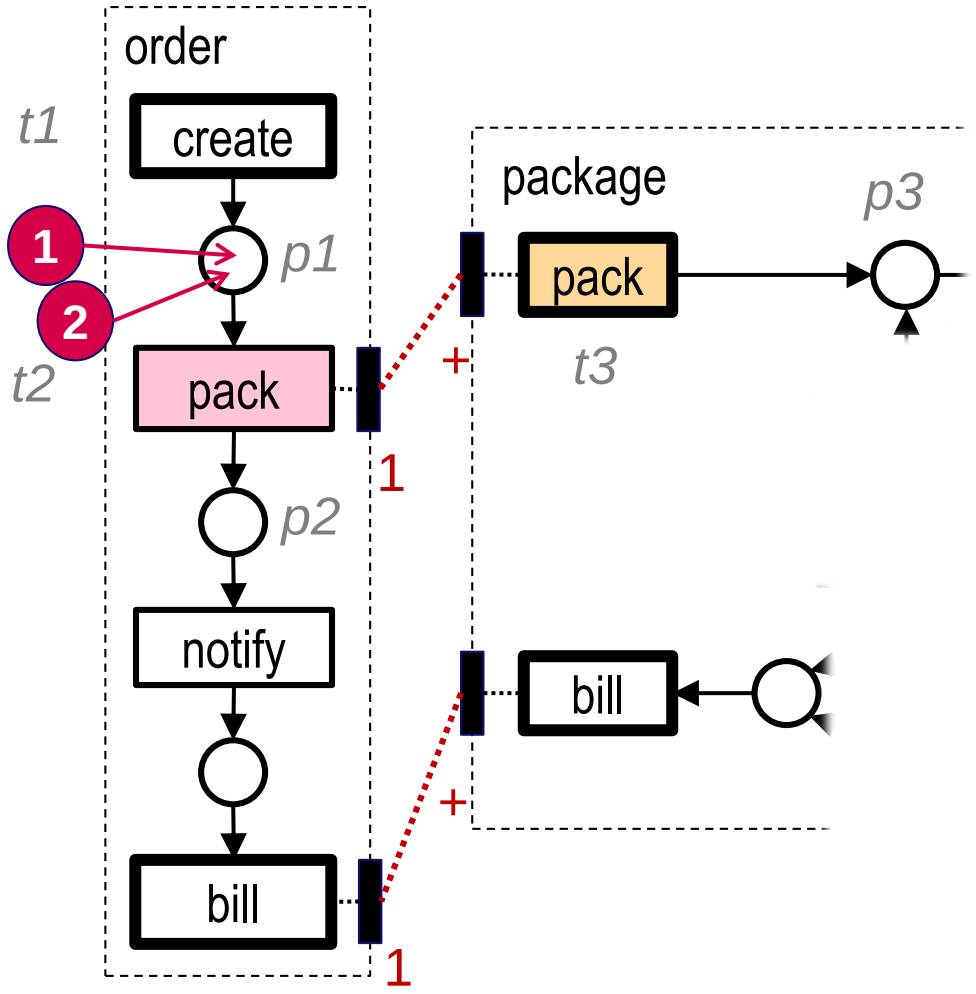
{ 1,21,25, 22, **2** }

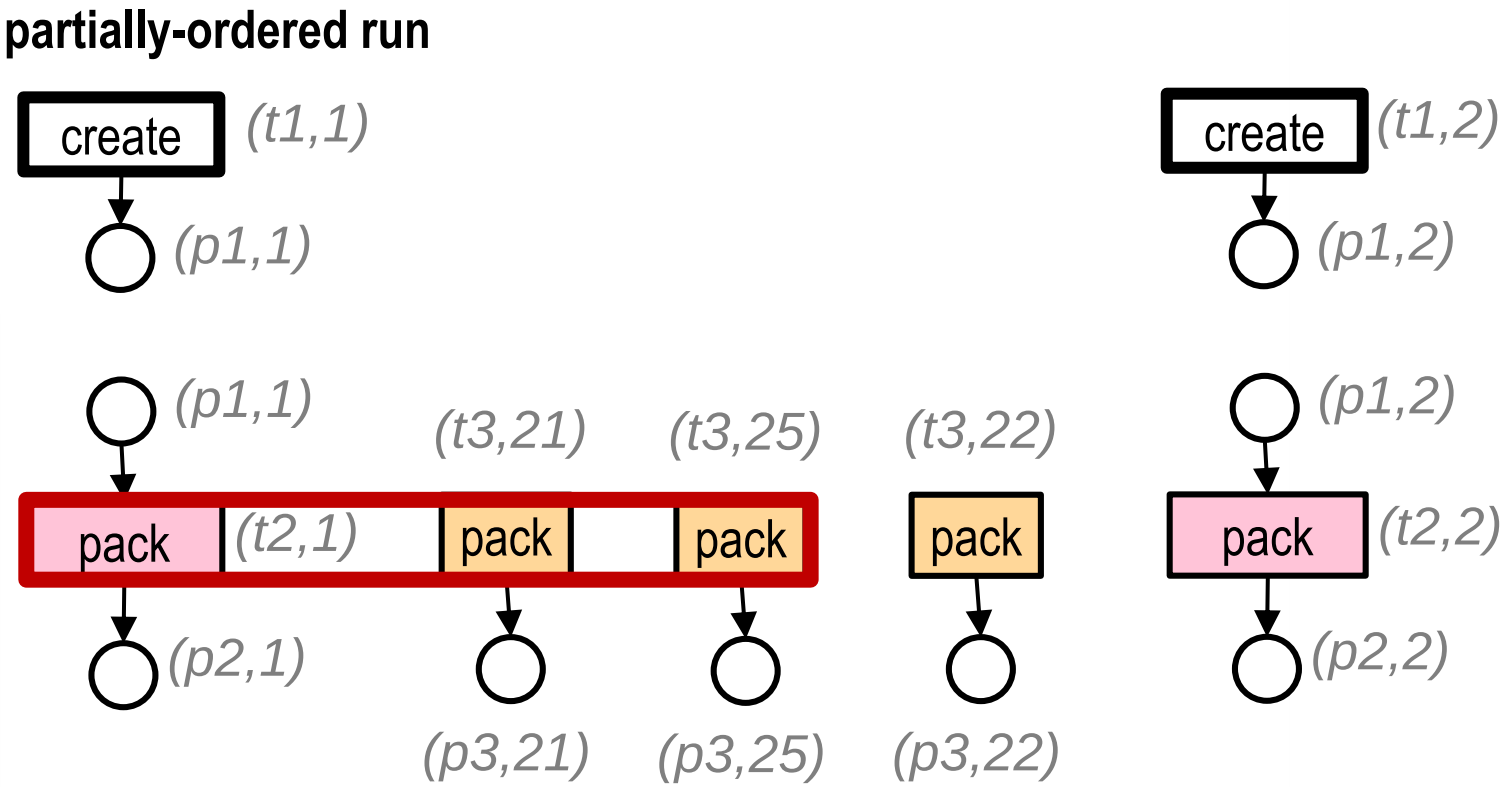
violates 1:+

***relation* underlying the binding**

{ (1,21),(1,25),(1,22),**(2,21),(2,25),(2,22)** }

Binding = non-det. chosen set of ids to sync



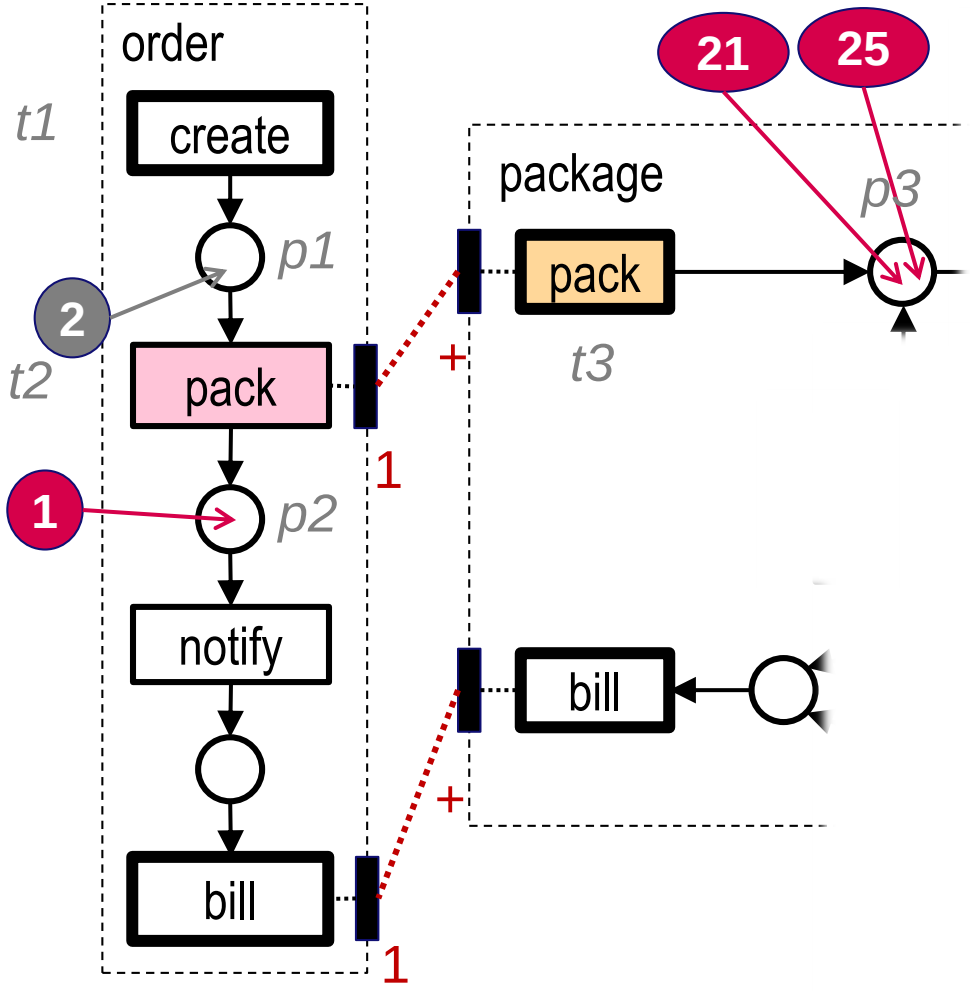


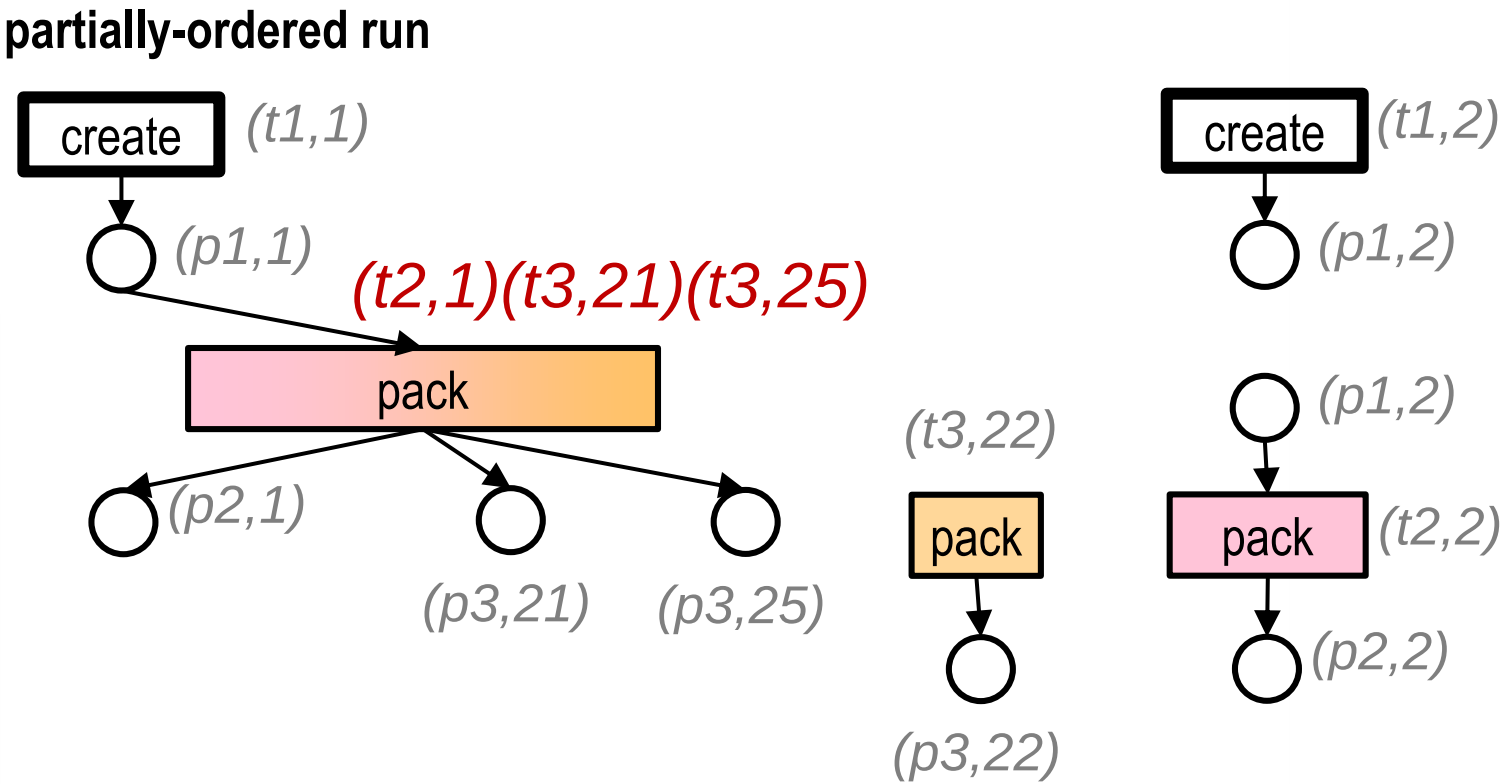
***binding* at channel (pack,pack):**
{ 1,21,25 }

***relation* underlying the binding**
{ (1,21),(1,25) }

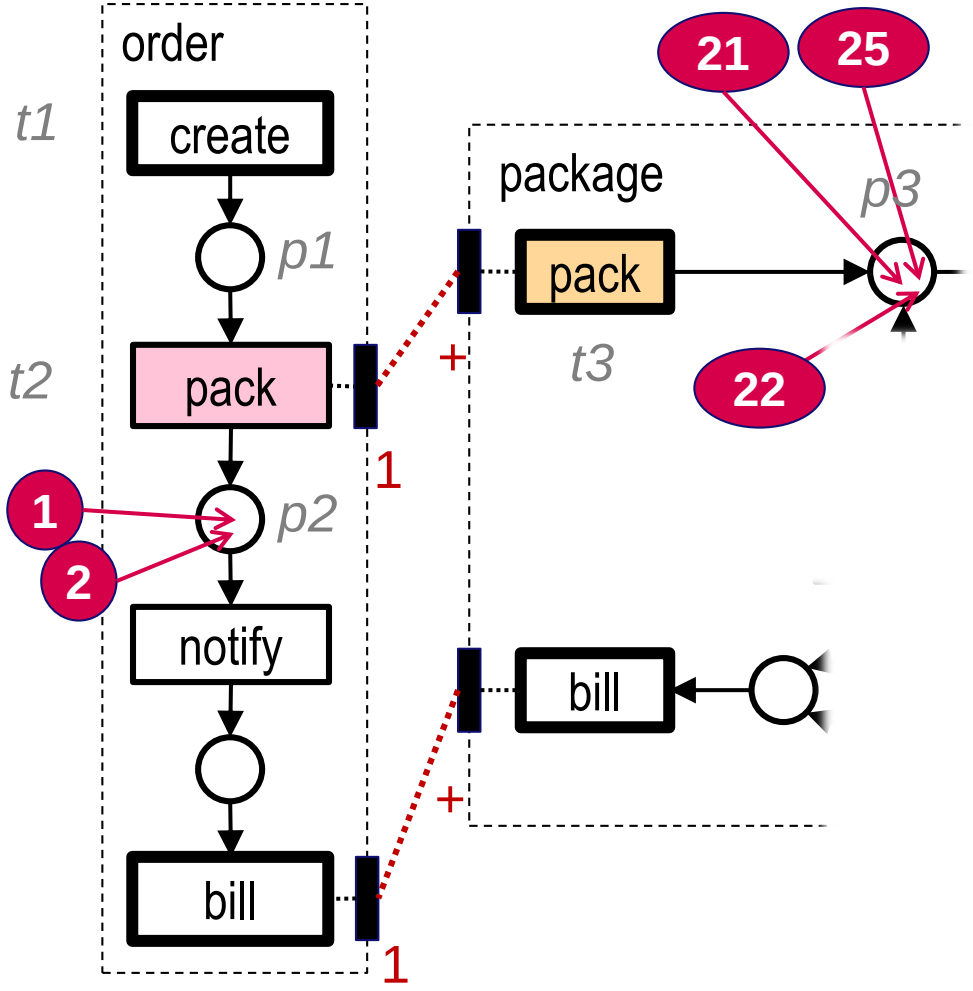
satisfies 1:+

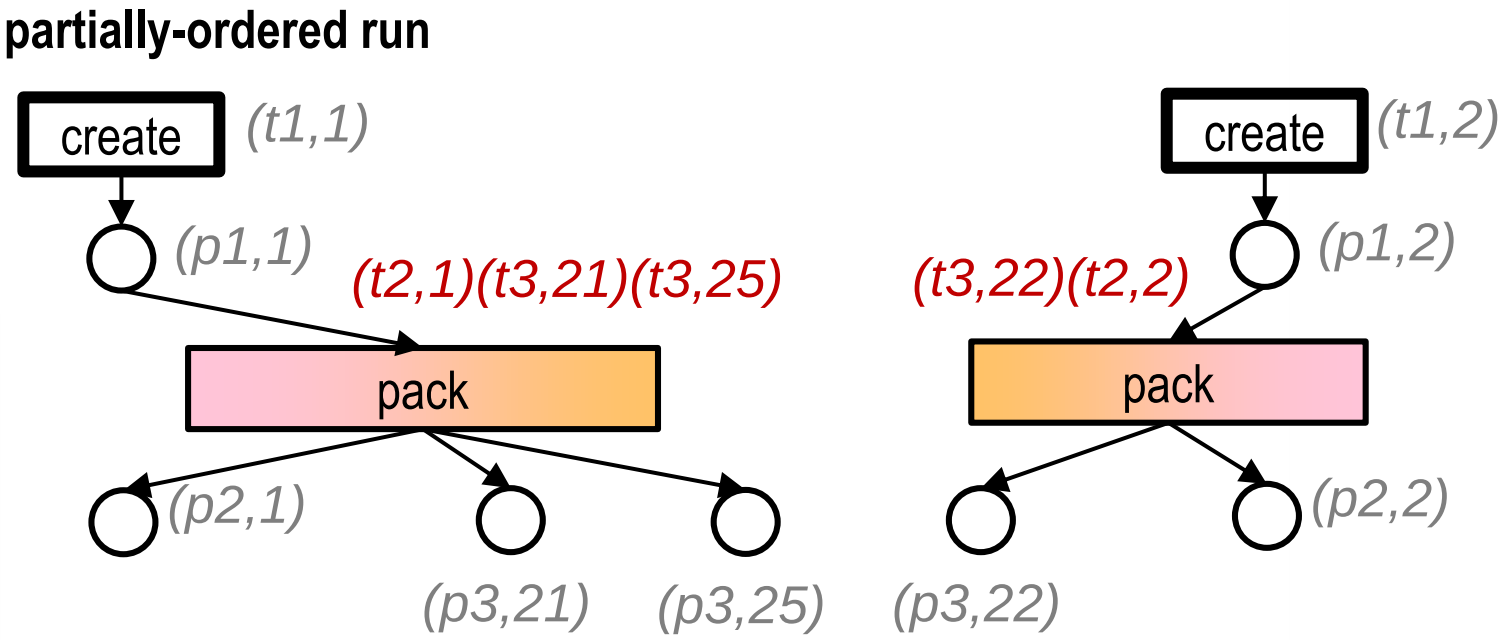
Unbounded synchronization of local events





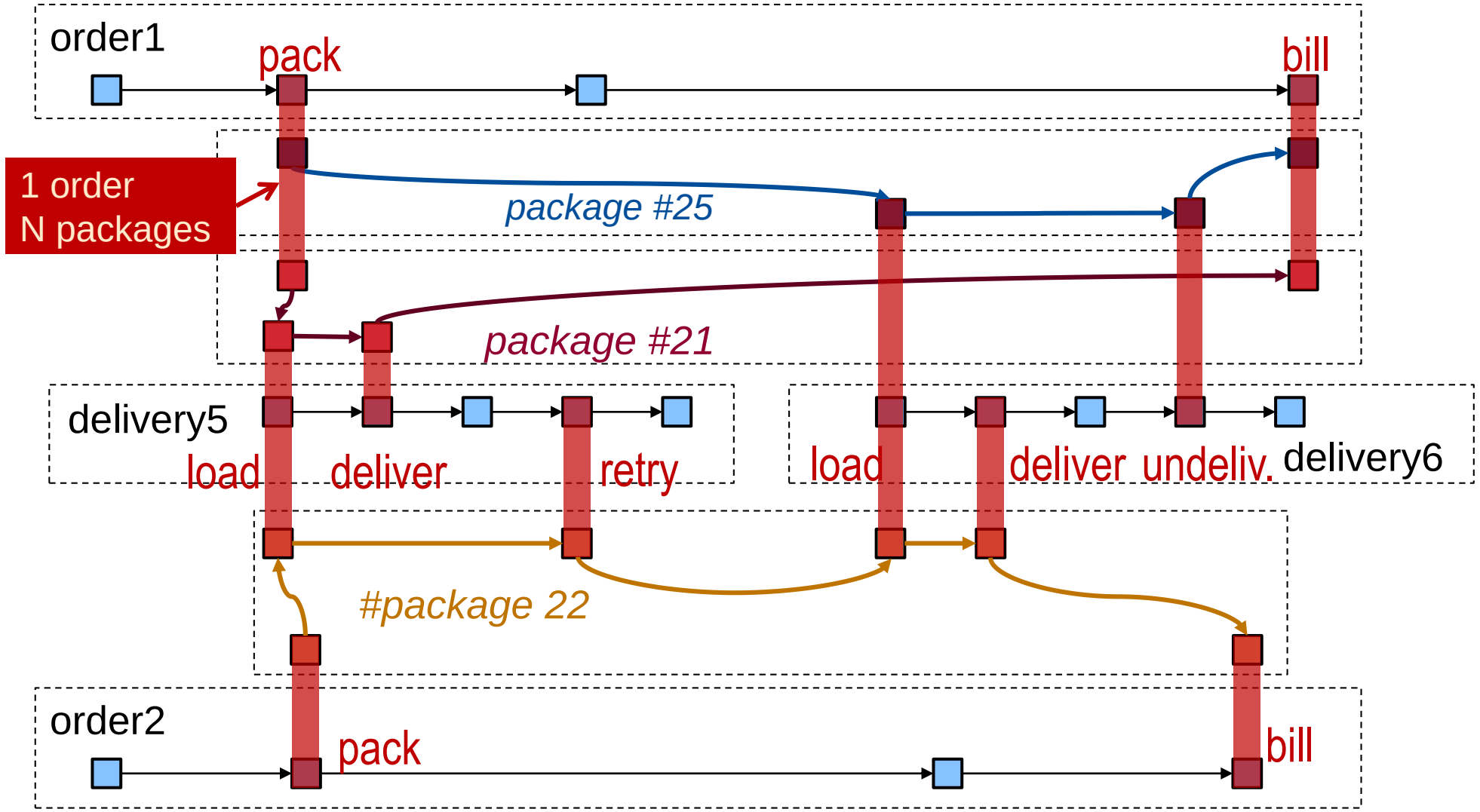
Unbounded synchronization of local events



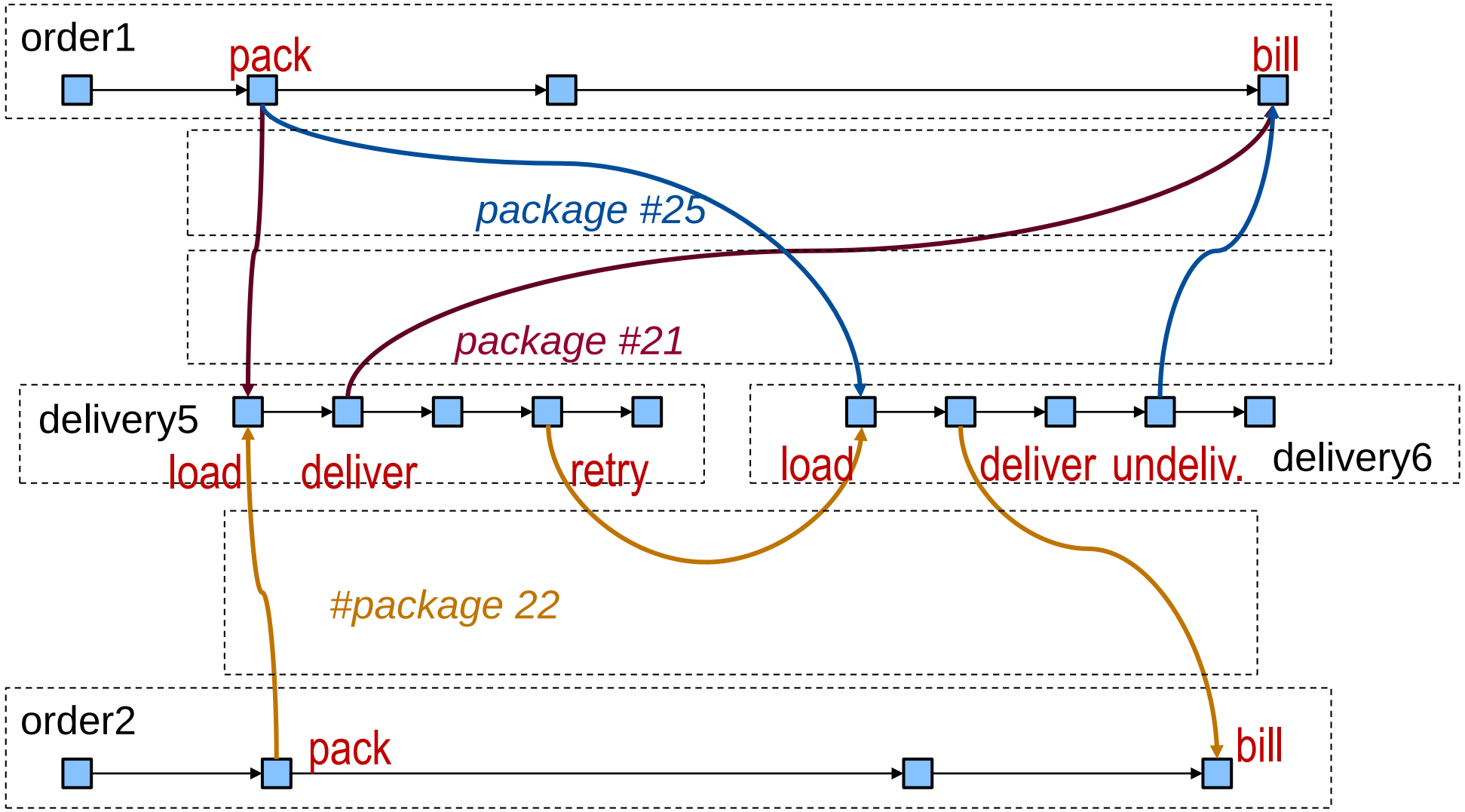


synchronization describes relation on behavioral level, **no data stored!**

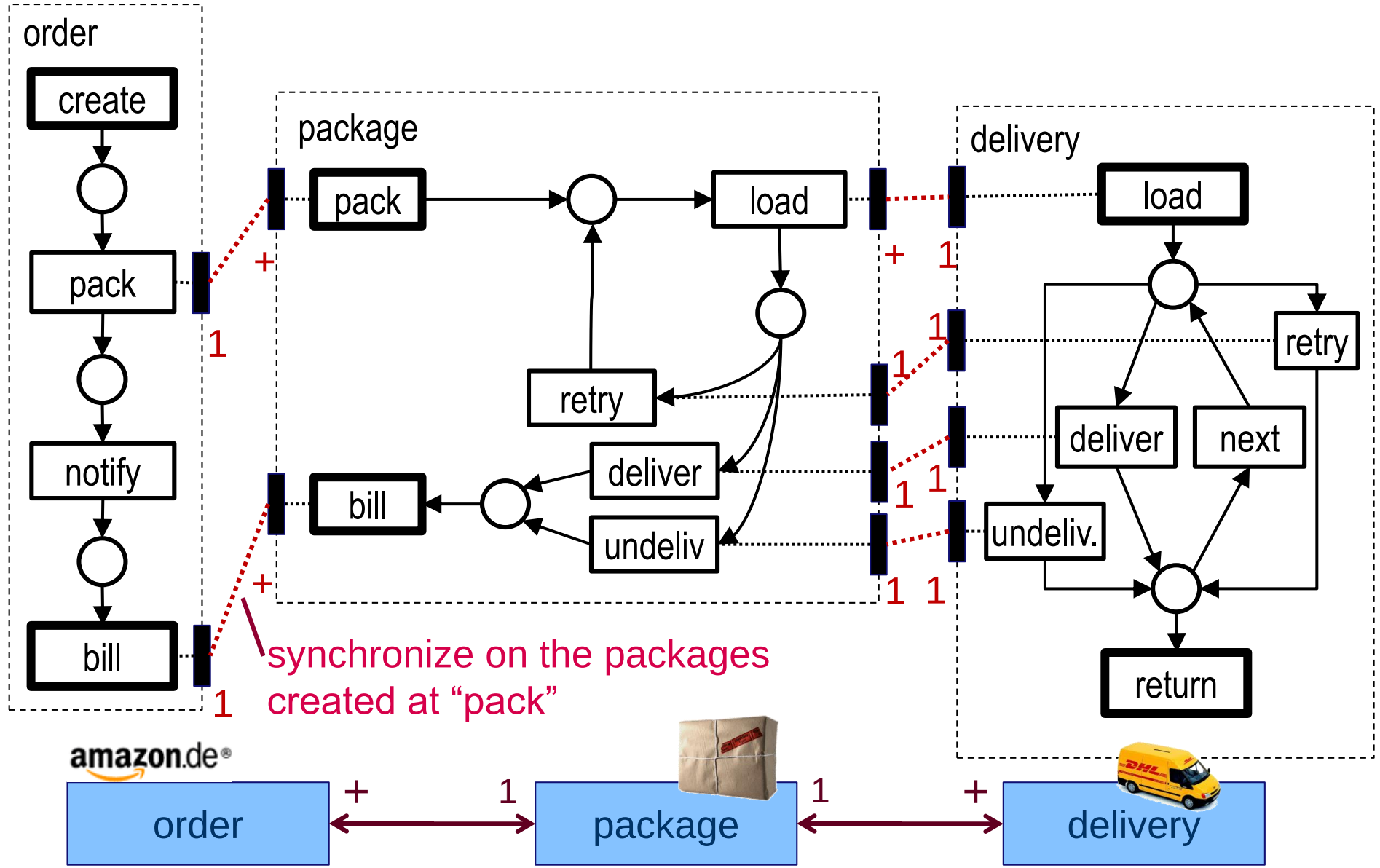
Each instance has 1 local run + synchronization



… compose into run with m:n interactions

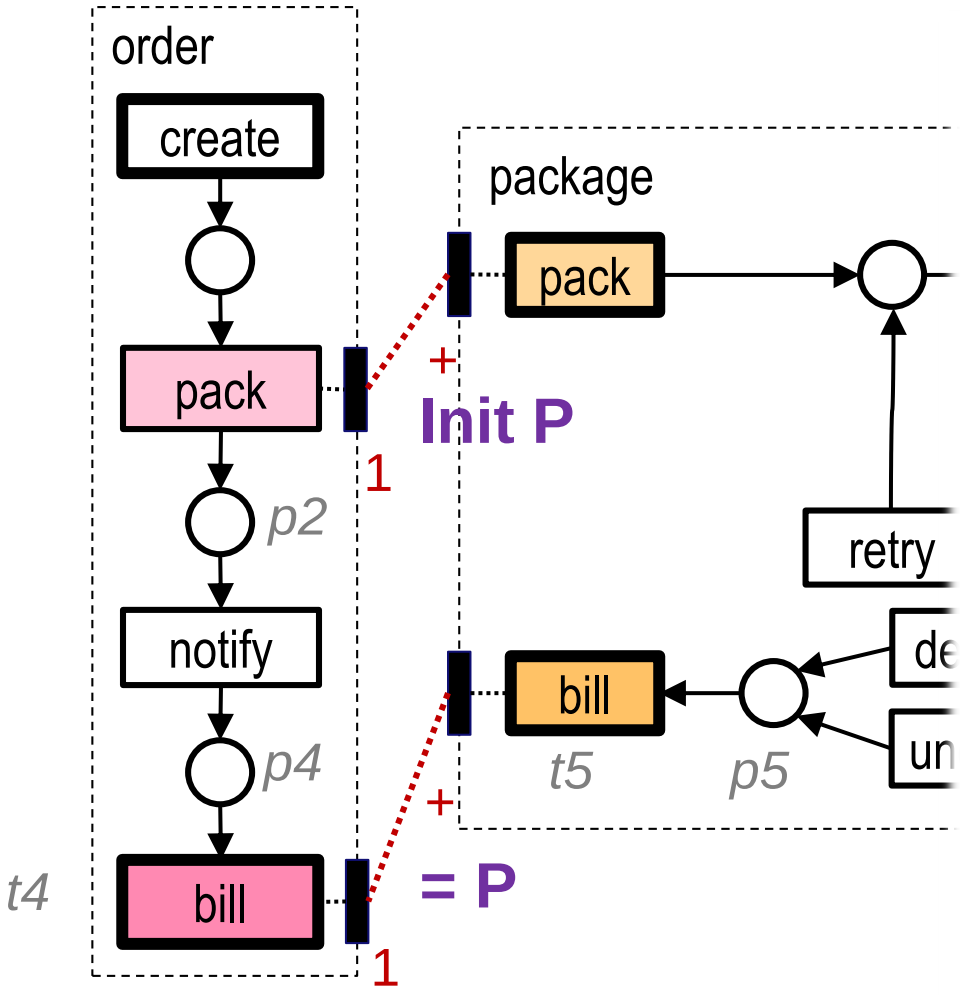


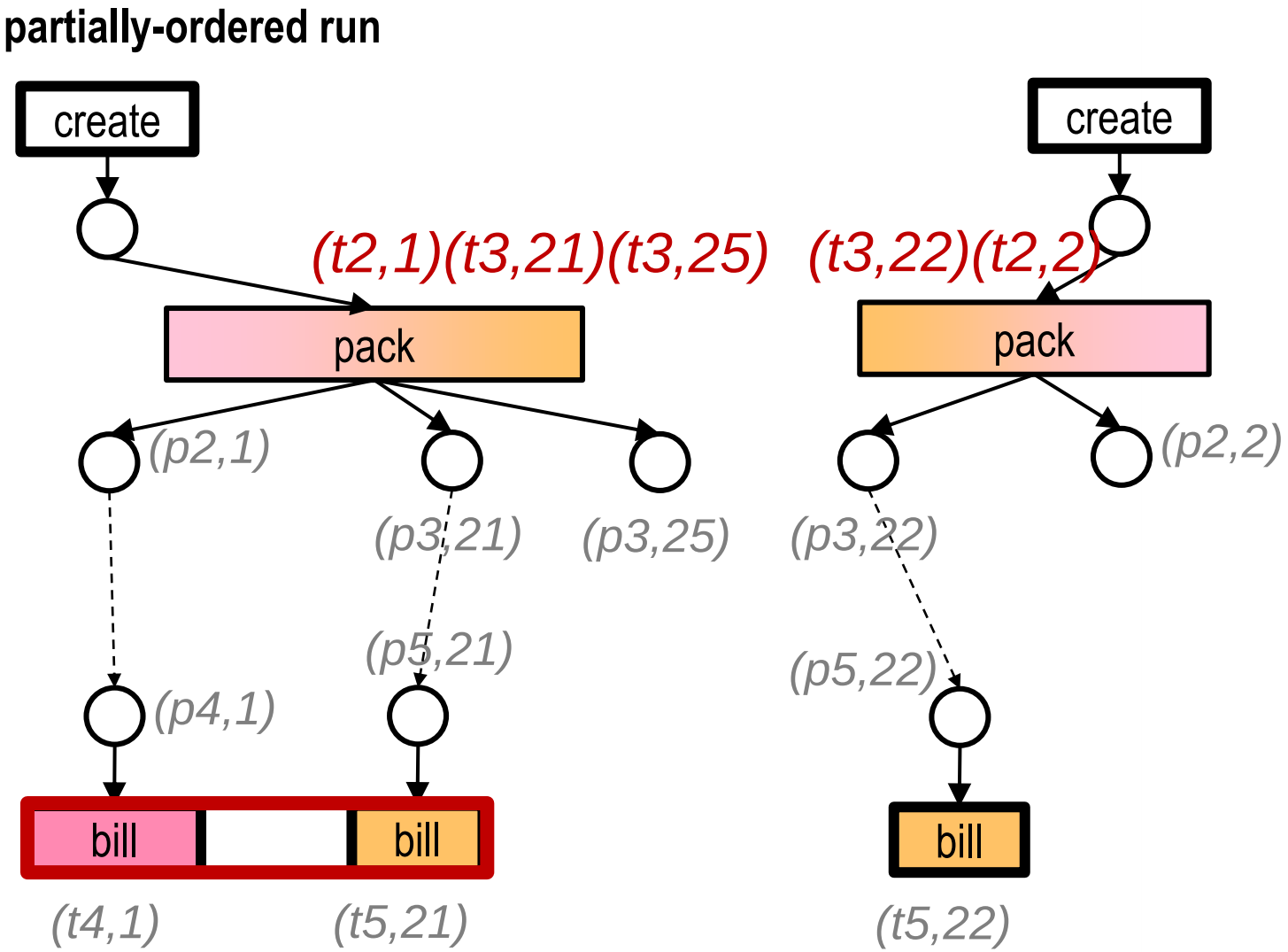
What's missing?



Initialize Correlation Binding P at “pack”

Match (=) Binding P at “bill”

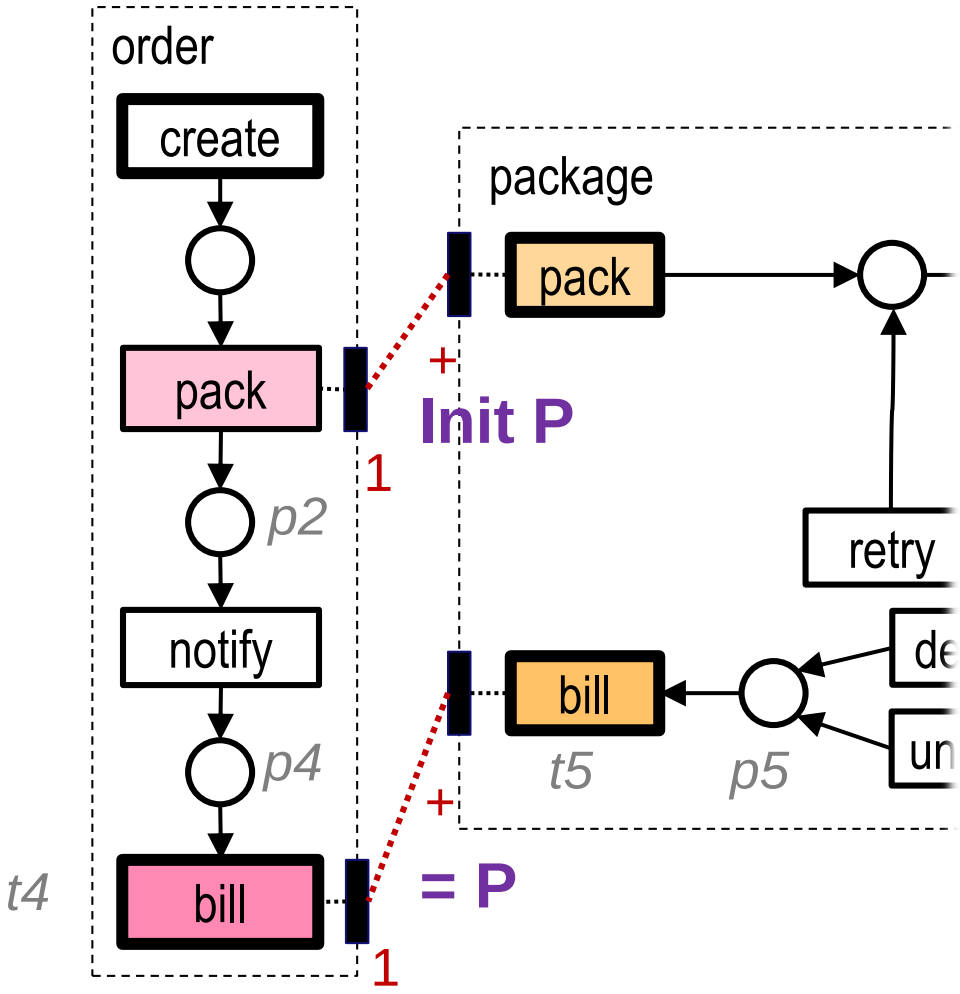


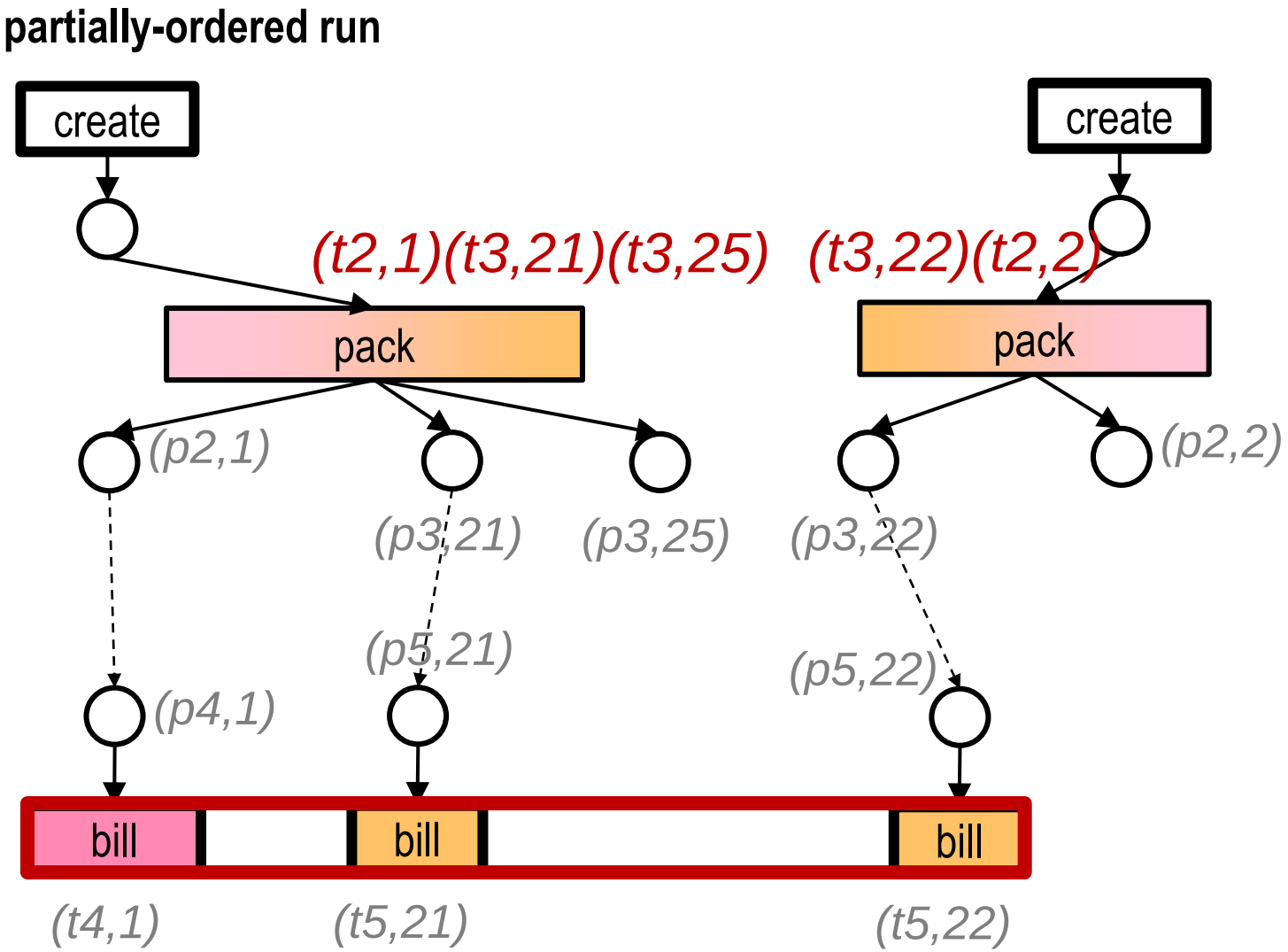


Init (1,21),(1,25) ≠ Match (1,21)

Initialize Correlation Binding P at “pack”

Match (=) Binding P at “bill”

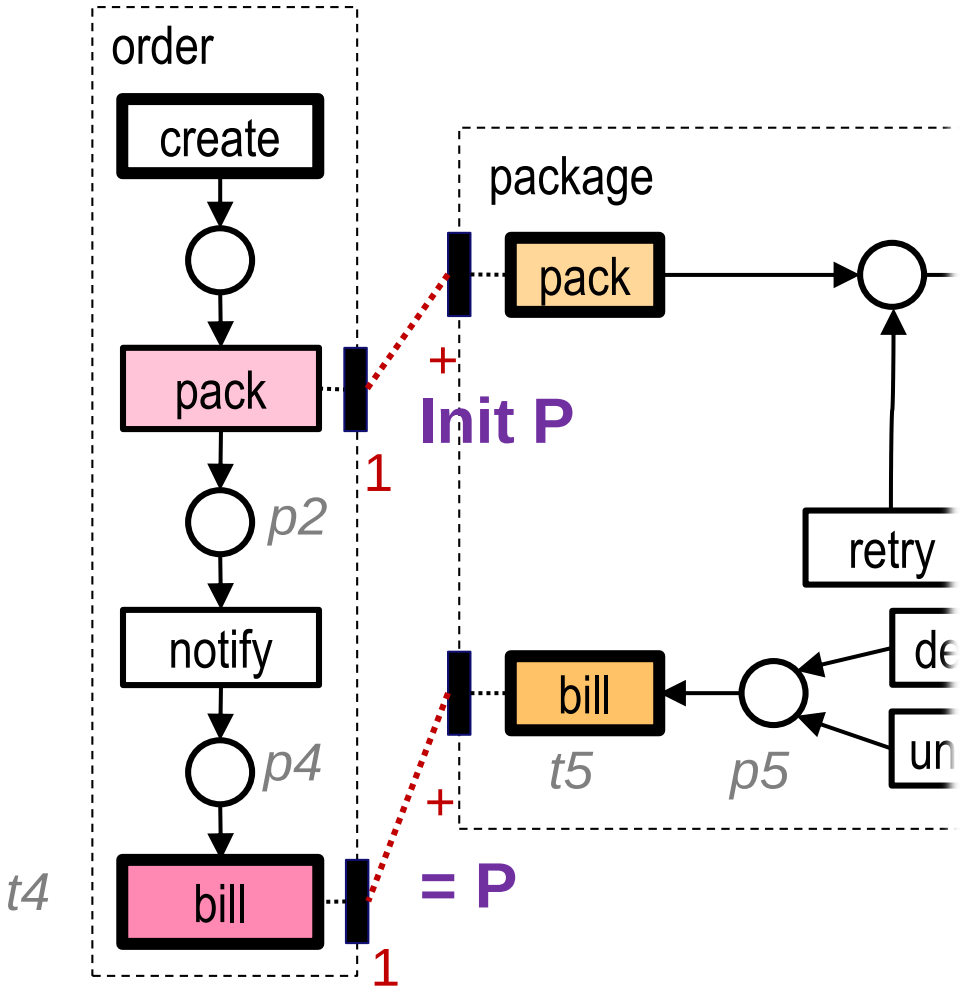


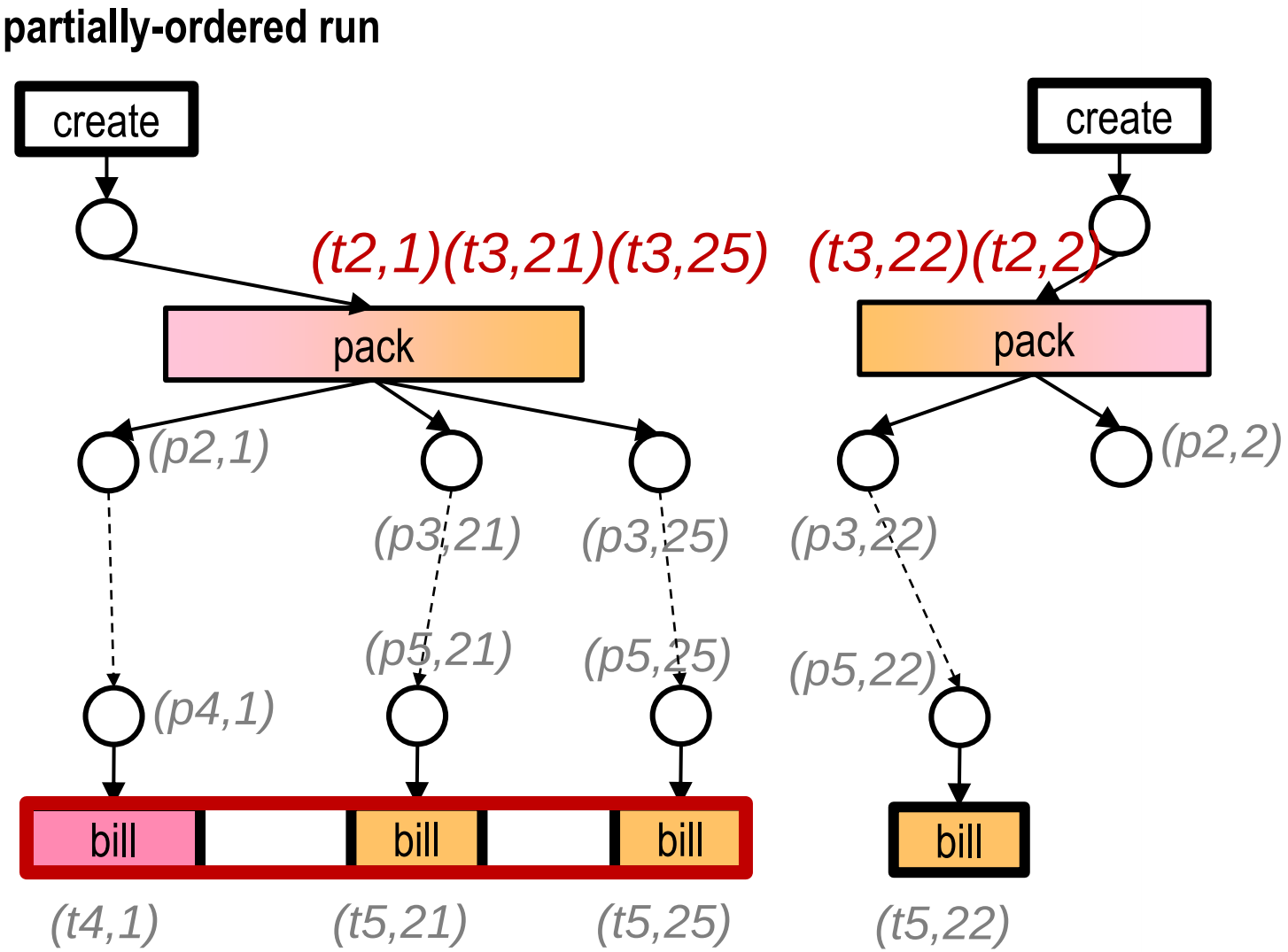


Init (1,21),(1,25) ≠ Match (1,21),(1,22)

Initialize Correlation Binding P at “pack”

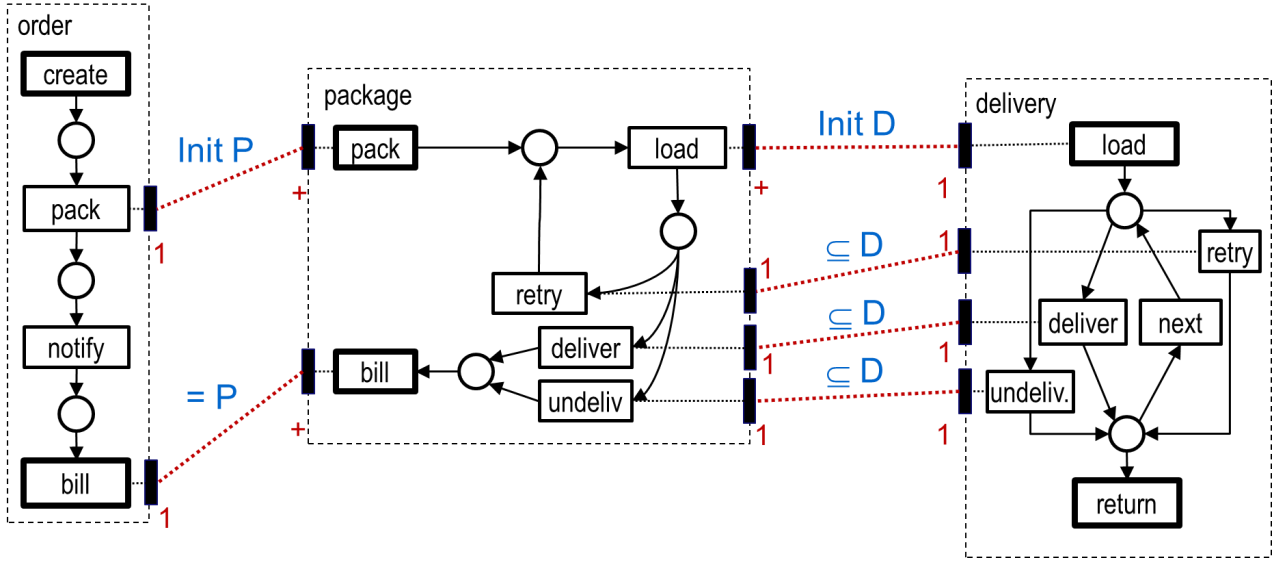
Match (=) Binding P at “bill”

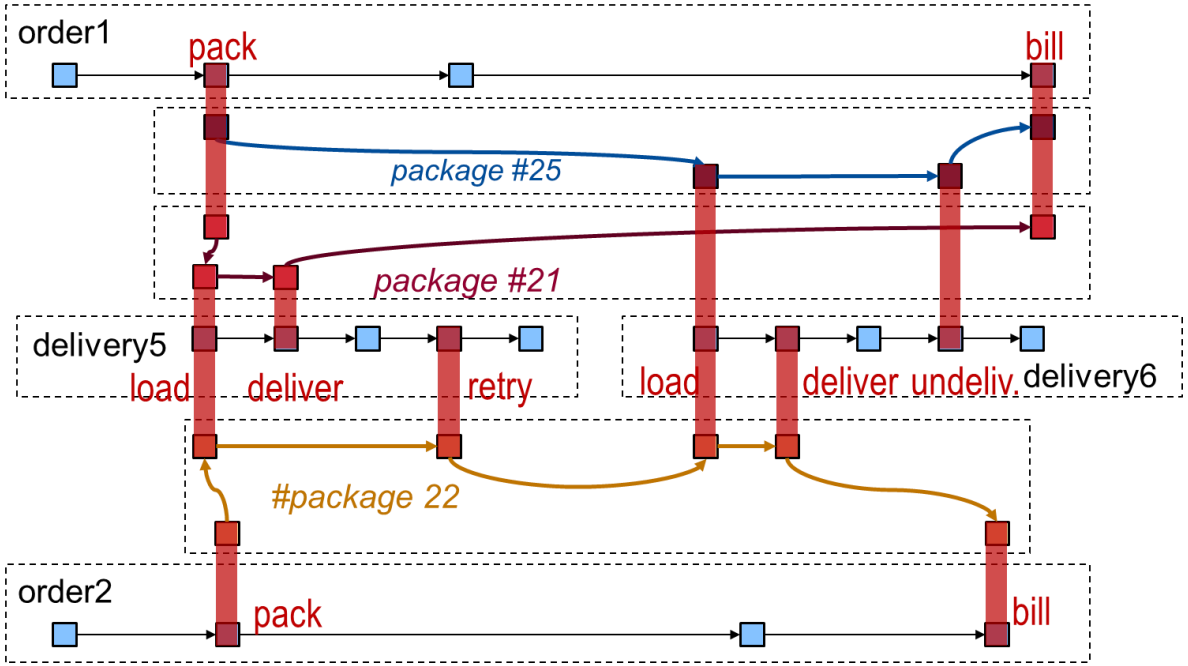




Init (1,21),(1,25) = Match (1,21),(1,25)

“Synchronous Proclets”





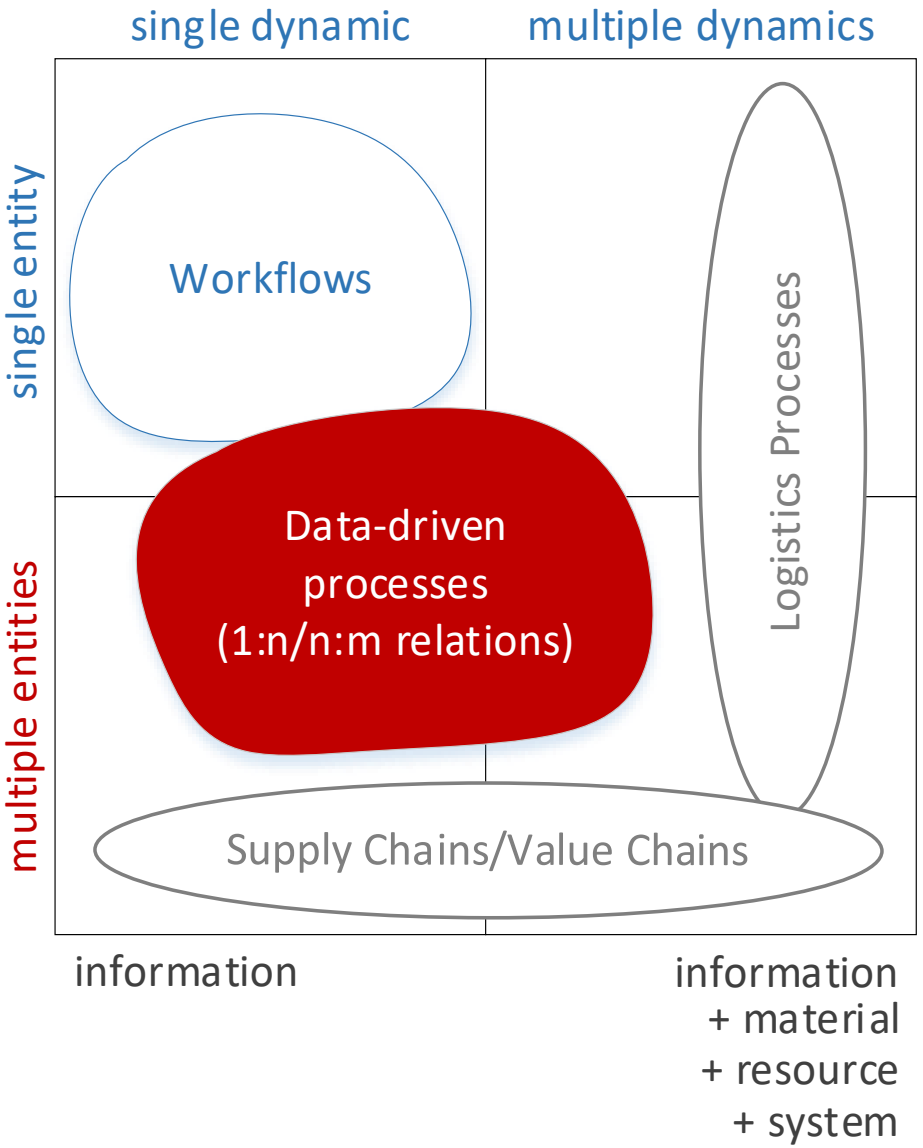
k process models

- Synchronous channels
- cardinality constraints 1:1, 1:n
- correlation constraints: Init P , ⊆P, =P

operational semantics: PO-runs

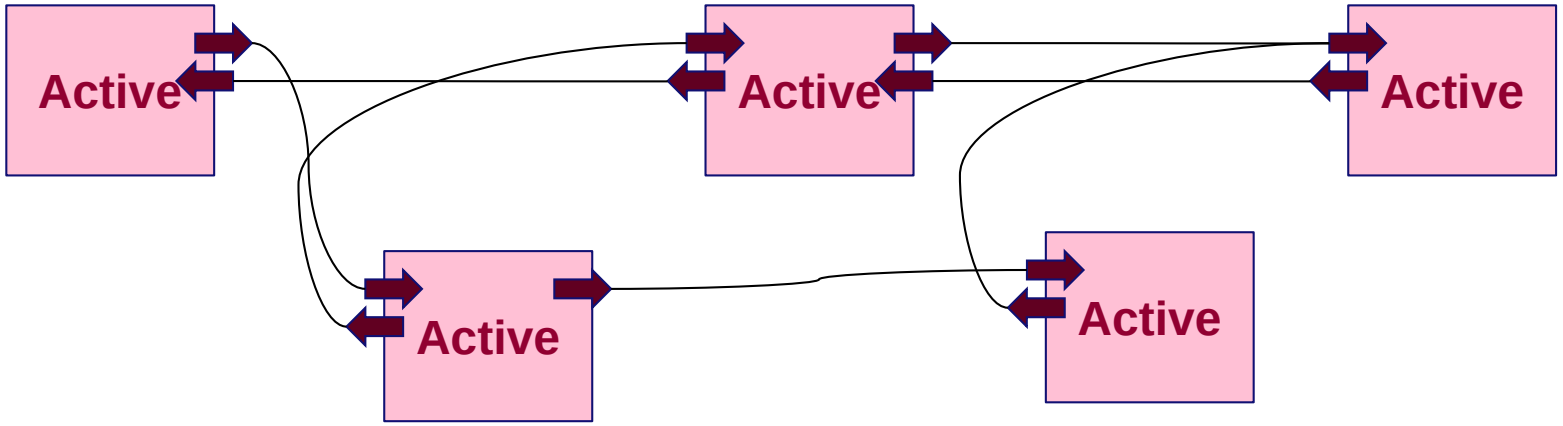
- id-tokens + ν-Petri net semantics
- unbounded synchronization of events
- events hold all relational information
- history-based correlation

Multi-Dimensional Processes



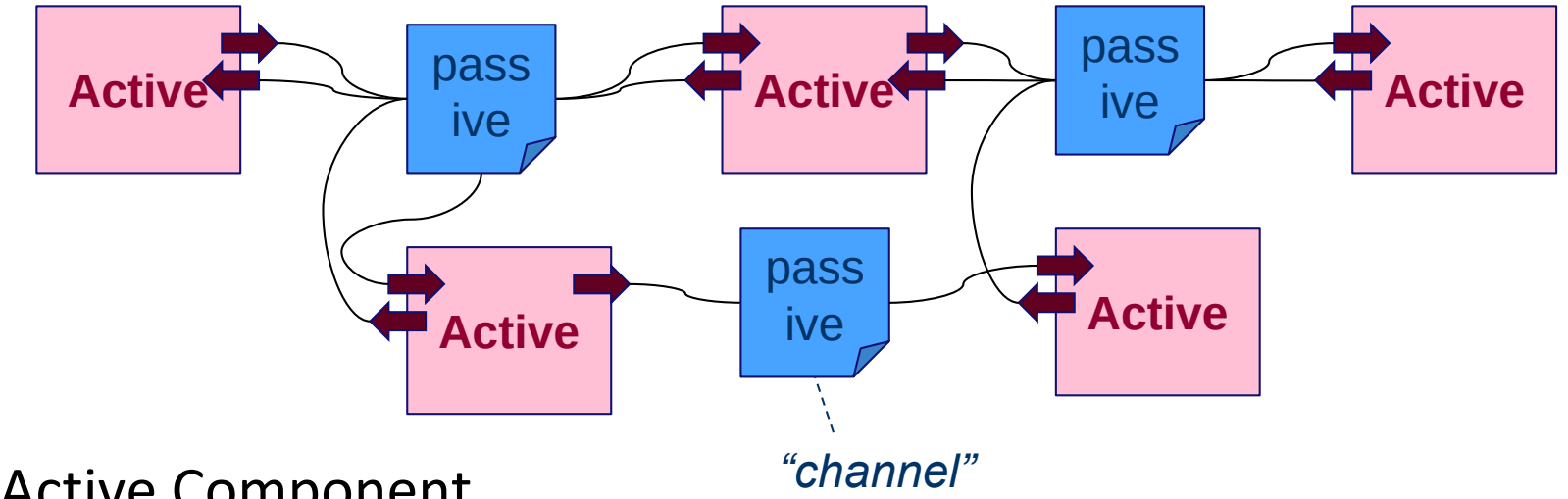
- Processes with multiple entities
  - How do they look like in practice?
  - Where does current modeling fail?
    - Implementation
    - Process mining
  - A proposal
  - **New research questions**

Active and Passive Components



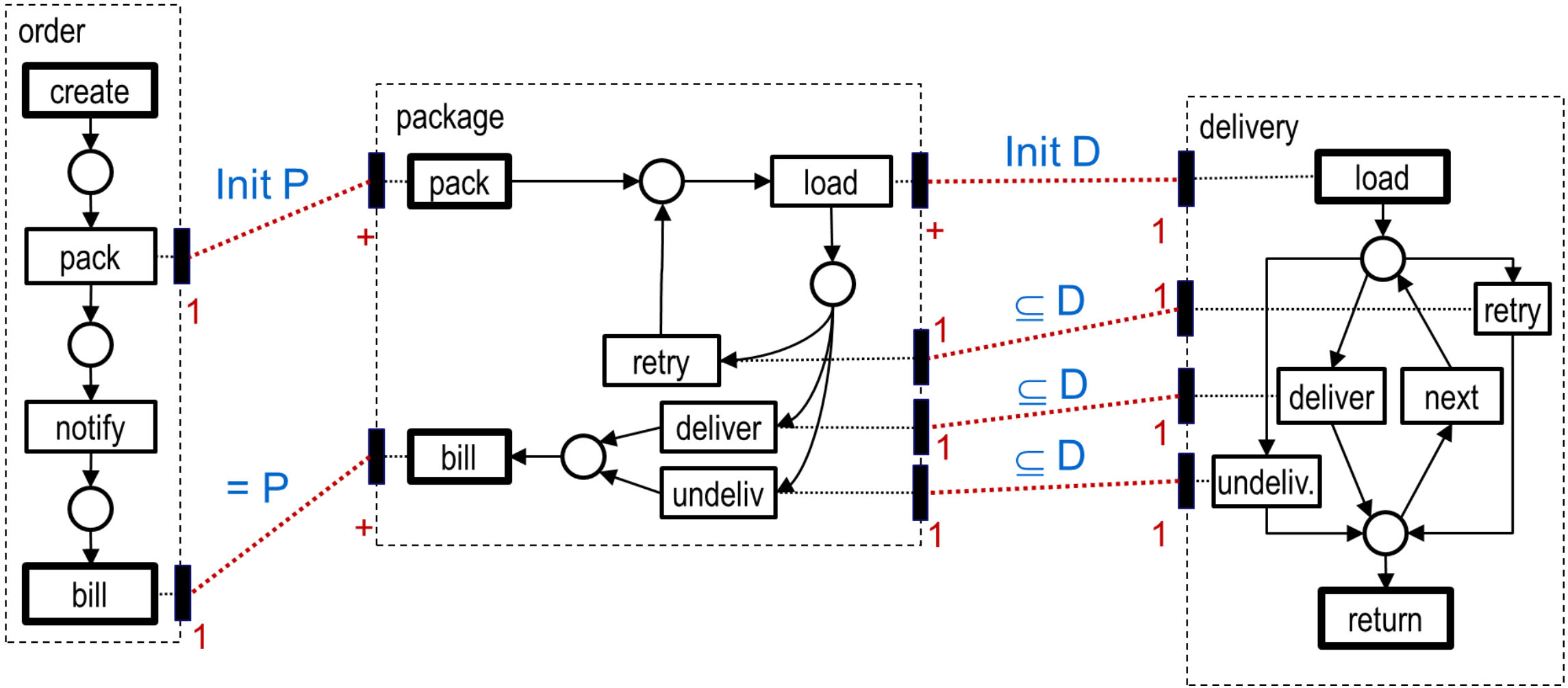
- Active Component
  - “service”: activities, performed by agents driving the process, e.g. split an order, write an invoice, …
  - can communicate (asynchronously)

Active and Passive Components



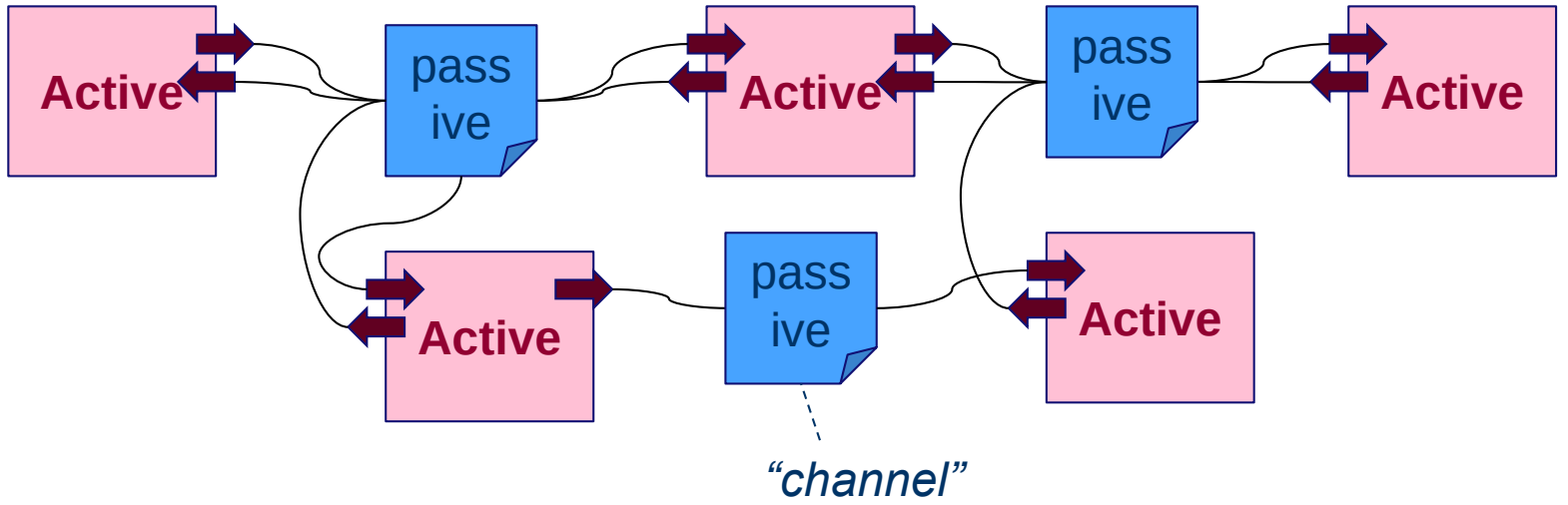
- Active Component
  - "process": activities, performed by agents driving the process, e.g. split an order, write an invoice, …
  - can communicate (asynchronously)
- Passive Component
  - "data object": attributes, can be updated, e.g. a package
  - order of updates is constrained, triggered from outside
  - **restricts interaction between several active processes**

Is this model sound?



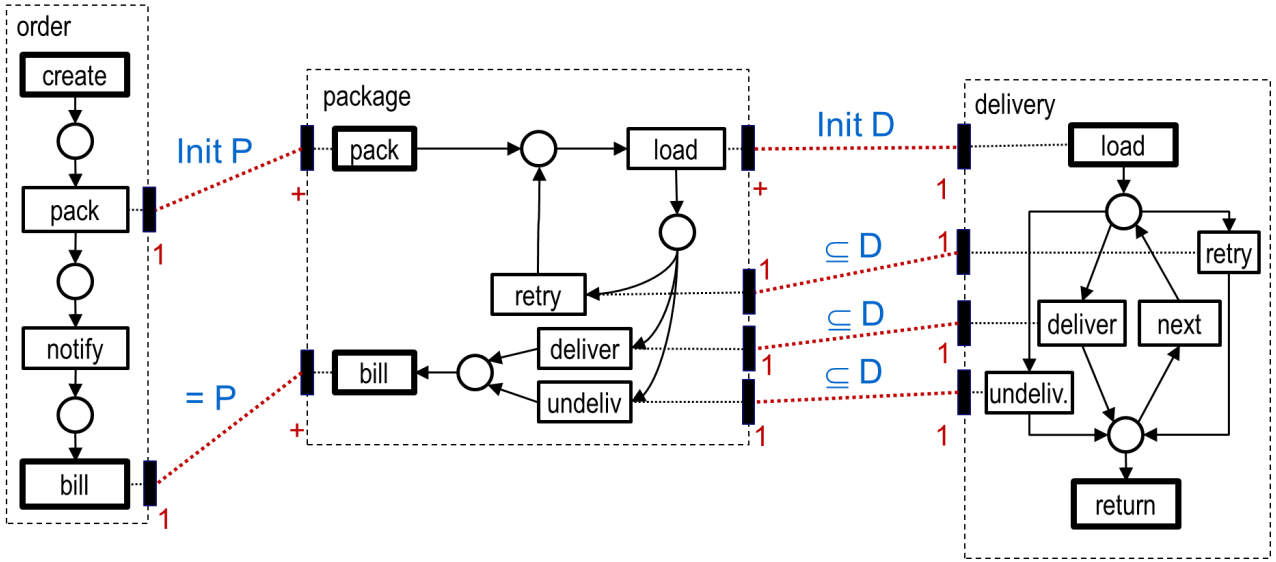
No, solution given in:
Dirk Fahland “Describing Behavior of Processes with Many-to-Many Interactions”, Petri Nets 2019

New verification & synthesis problems



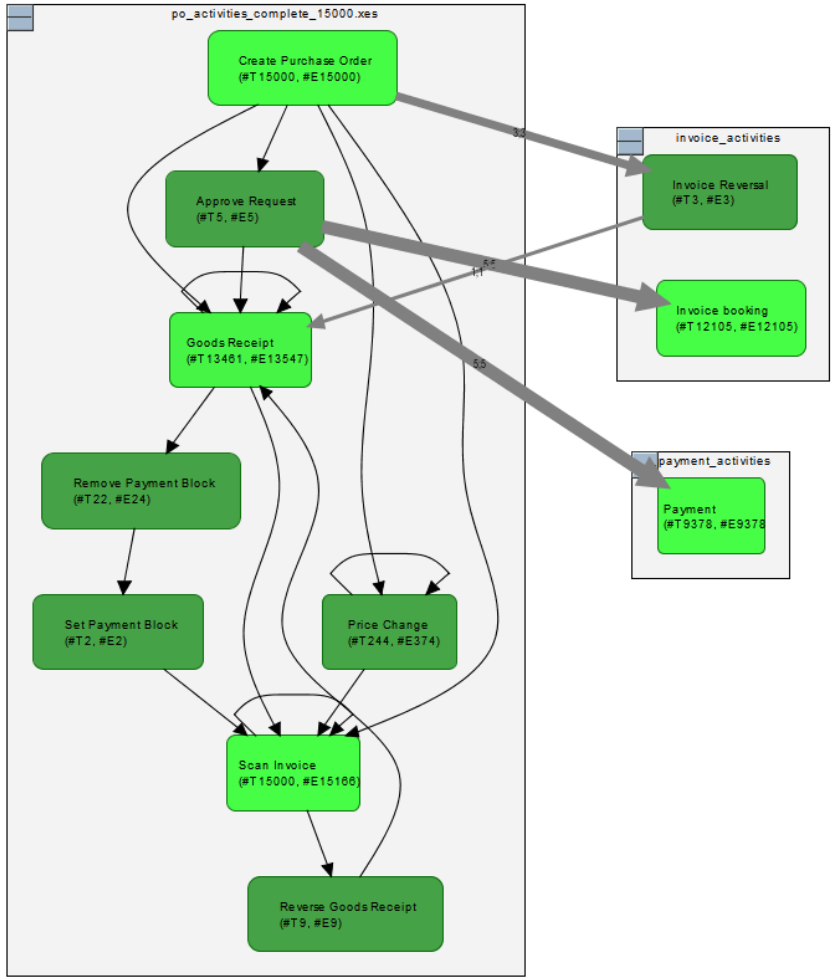
- Is reachability **decidable**?
- **Is the model „sound“?**
  → parameterized verification problem (for all number of instances)
- What‘s the **minimal number of instances** needed to achieve a particular (problematic) behavior?
- Given k „active“ components
  → **synthesize the „passive“ components** to ensure property X

New Modeling (Language) Problems

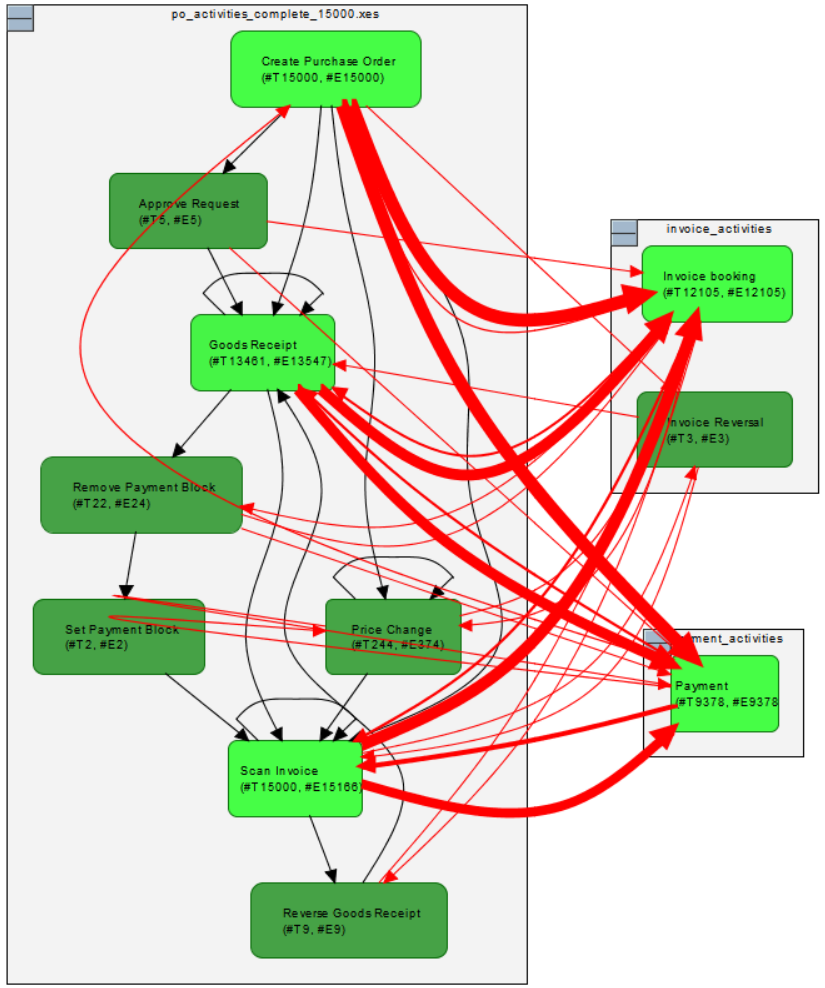


- How to add **determinism**, e.g., link to **data model, guard conditions**
  - Data-Centric Dynamic Systems, DB-Petri Nets (Marco Montali)
- What are **adequate design patterns?**
- How to design **comprehensible models** (for task X)**?**
  - These models flatten the 1:n/n:m interactions
  - Strong “visual” discrepancy between model syntax and behavior

Problem: Flattened n:m relations & variants

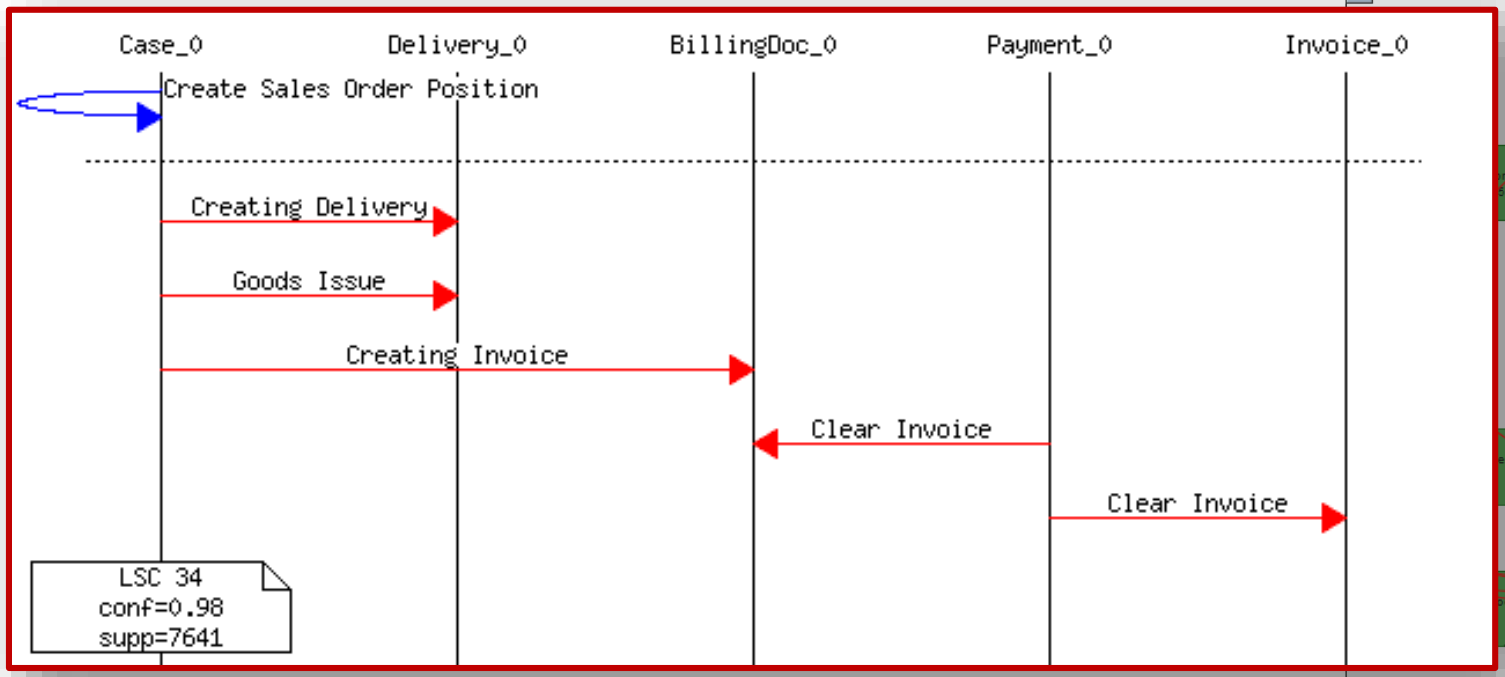


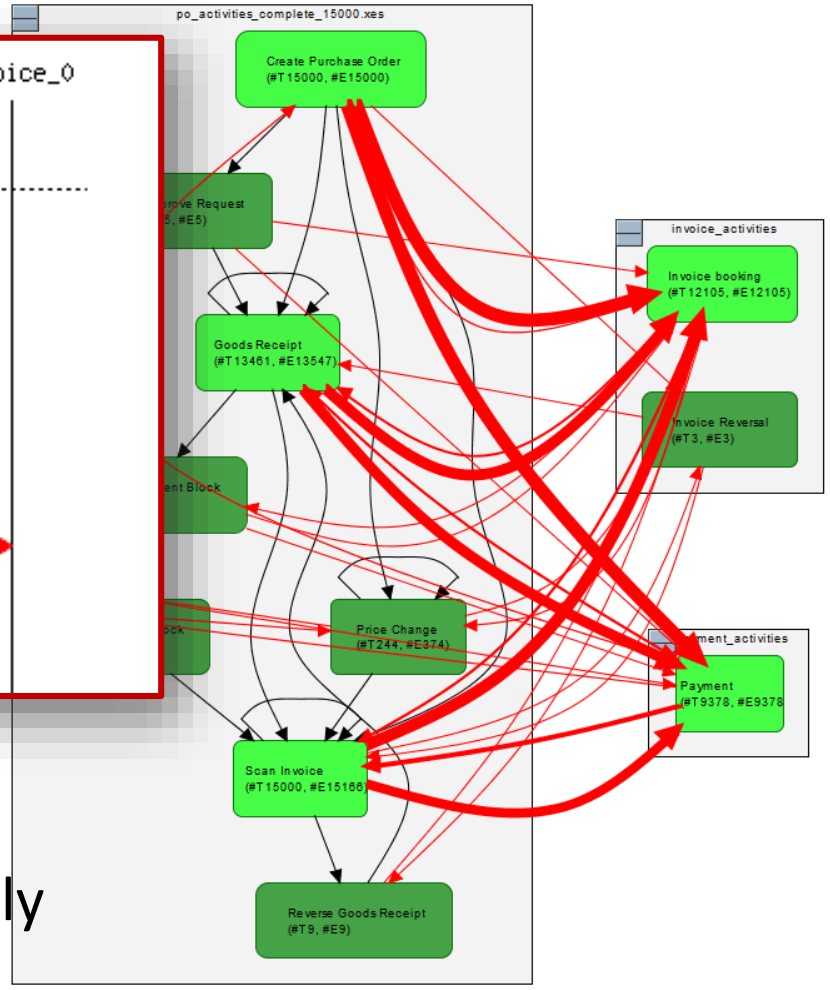
most frequent behavior



all behaviors across 3 data objects

Explicitly describing behavior over n:m relations

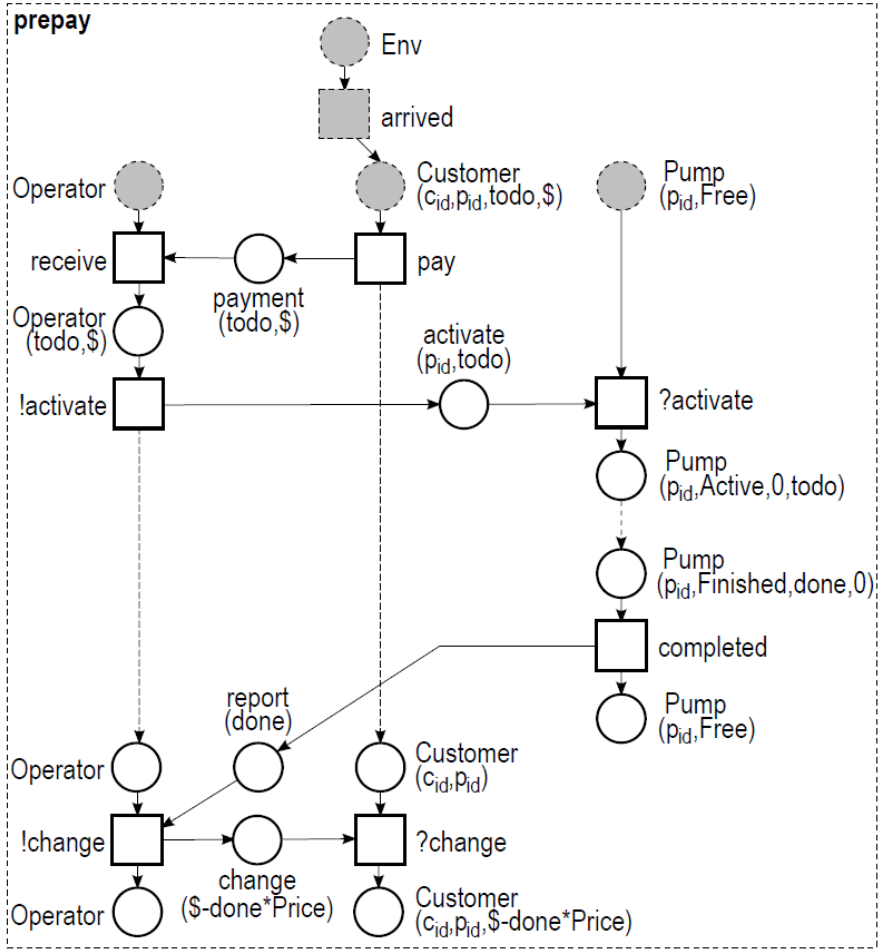


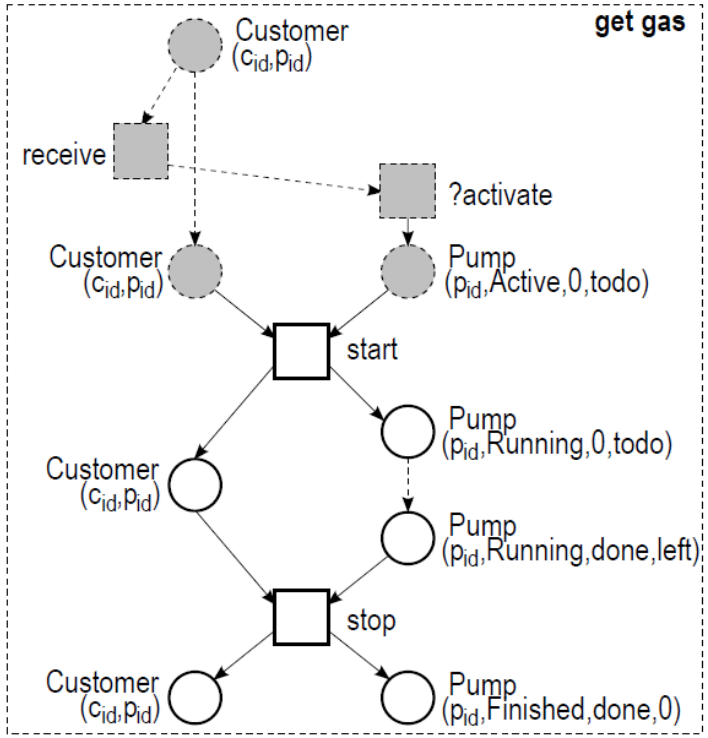


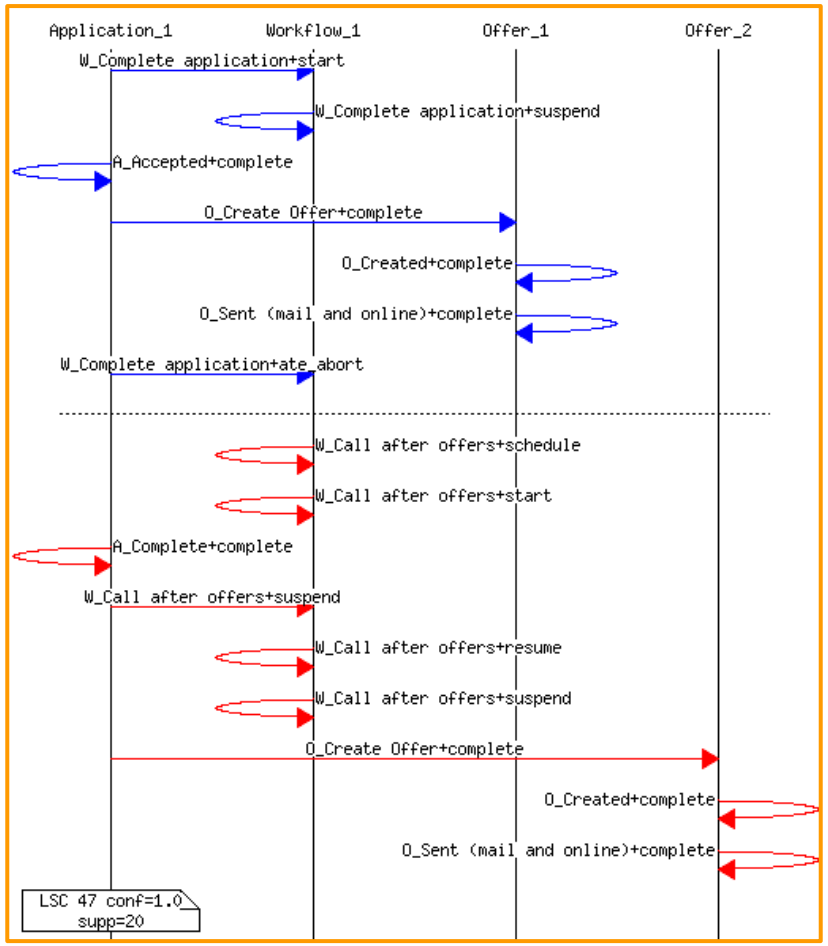
Scenario:

- Describe partial behavior
- Describe object instances + interactions explicitly
- + context (e.g., enabling history)

Scenarios w/ history specify m:n interactions in a visual, comprehensive way



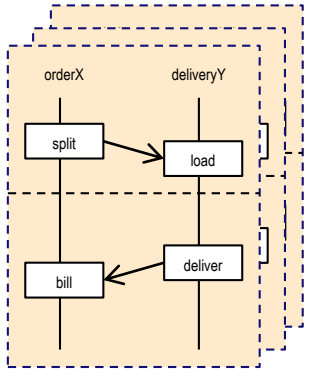




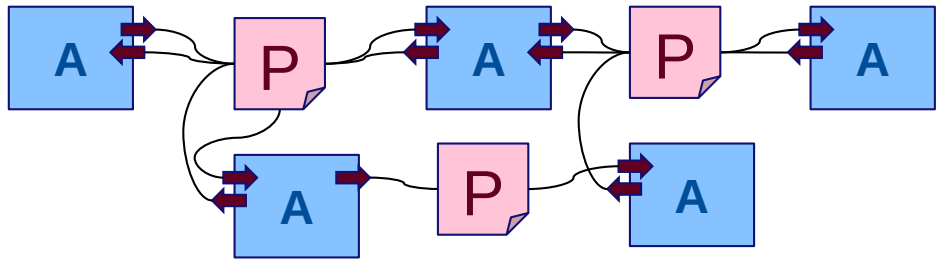
Dirk Fahland “Oclets - Scenario-Based Modeling with Petri Nets.” Petri Nets 2009: 223-242

Dirk Fahland, Robert Prüfer: Data and Abstraction for Scenario-Based Modeling with Petri Nets. Petri Nets 2012: 168-187

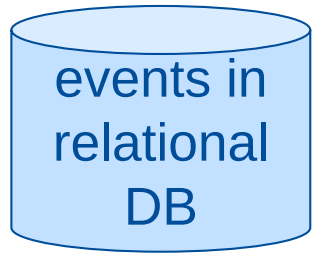
Need to transform between local/global views

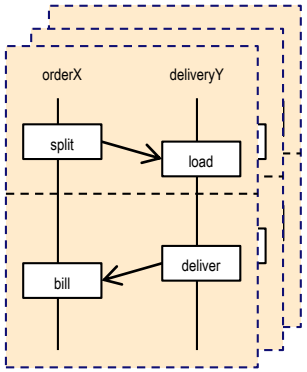


scenarios between objects

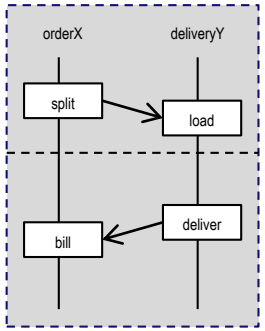


process model of processes + data objects

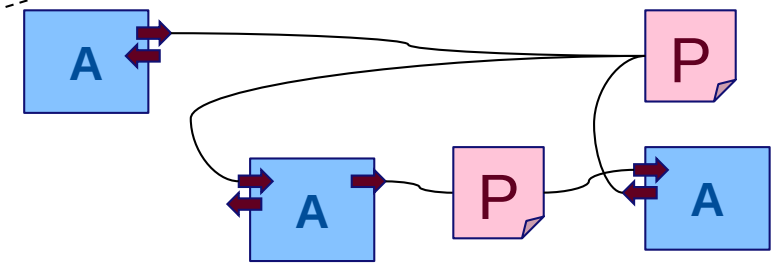




all behavior/view



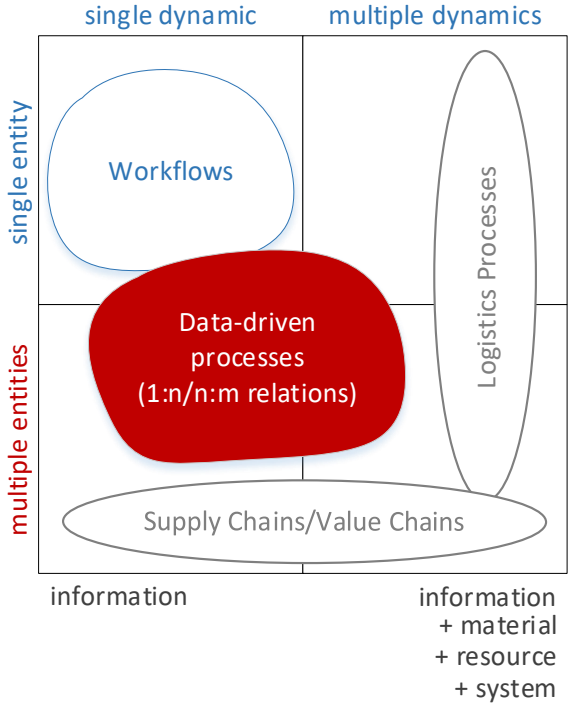
behavior of interest

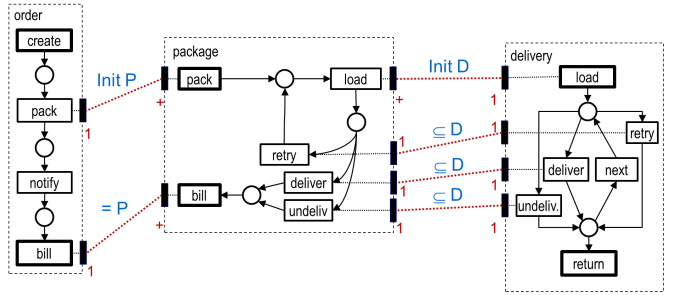


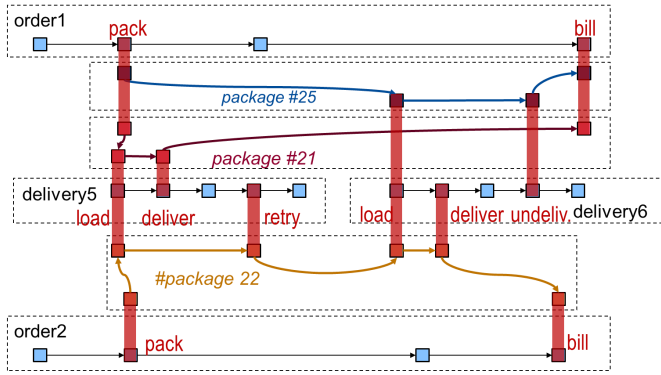
view

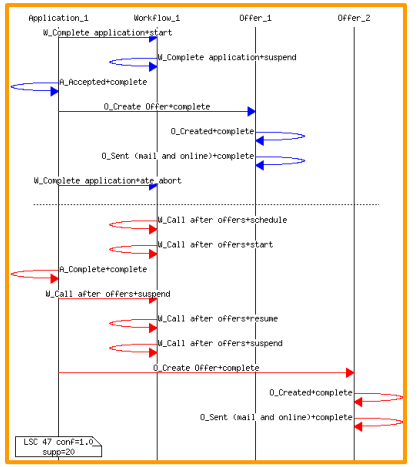
Wrap-Up

- **Multi-Dimensional Process Dynamics**
- **Net-based model**
  - partially ordered runs
  - Id-tokens
  - Unbounded synchronization at events
  - History-based correlation
- **Gives rise to many new research questions**
  - Decidability, verification
  - Scenario-based modeling w/ history
  - Synthesis, querying









Reading

- Dirk Fahland, Massimiliano de Leoni, Boudewijn F. van Dongen, Wil M. P. van der Aalst:
  "Many-to-Many: Some Observations on Interactions in Artifact Choreographies." ZEUS 2011: 9-15
  The running example of orders and deliveries, and the idea of synchronous interactions of artifacts with processes.
- Dirk Fahland **"Describing Behavior of Processes with Many-to-Many Interactions"**, Petri Nets 2019
- Dirk Fahland **"Oclets - Scenario-Based Modeling with Petri Nets."** Petri Nets 2009: 223-242
- Dirk Fahland, Robert Prüfer: **Data and Abstraction for Scenario-Based Modeling with Petri Nets.** Petri Nets 2012: 168-187
- W.M.P. van der Aalst, R.S. Mans, and N.C. Russell. "**Workflow Support Using Proclets: Divide Interact, and Conquer**." IEEE Data Eng. Bull., 32(3):16-22, 2009
- A. Nigam and N. Caswell, "**Business artifacts: An approach to operational specification,**" IBM Systems Journal, vol. 42, no. 3, pp. 428–445, 2003.
- D. Cohn and R. Hull, "**Business artifacts: A data-centric approach to modeling business operations and processes**" Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, vol. 32, no. 3, pp. 3–9, 2009
- ACSI Project "**The core ACSI artifact paradigm: artifact-layer and realization-layer**"
  http://www.acsi-project.eu/deliverables/D1.1_The_core_ACSI_artifact_paradigm.pdf
  Different models for Artifact-Centric Processes (FSM, GSM, Proclets)
- E. Nooijen, B. v. Dongen, and D. Fahland, "**Automatic Discovery of Data-Centric and Artifact-Centric Processes**," in Business Process Management Workshops. Springer, 2013, pp. 316–327.
  Discovering Petri net life-cycle models from databases
- Viara Popova, Dirk Fahland, Marlon Dumas: "**Artifact Lifecycle Discovery."** Int. J. Cooperative Inf. Syst. 24(1) (2015)
  Discovering GSM models from raw logs
- Xixi Lu, Marijn Nagelkerke, Dirk Fahland: "**Discovering Interacting Artifacts from ERP systems**" IEEE Trans. On Services Computing (to appear), http://dx.doi.org/10.1109/TSC.2015.2474358
  Discovering Proclet models with interactions from databases
- Andreas Meyer, Luise Pufahl, Kimon Batoulis, Dirk Fahland, Mathias Weske: "**Automating data exchange in process choreographies**". Inf. Syst. 53: 296-329 (2015)
  Extending BPMN data objects with annotations to automated data dependencies, message creation, message correlation, message processing