Integrated Modelling and Verification of Processes and Data

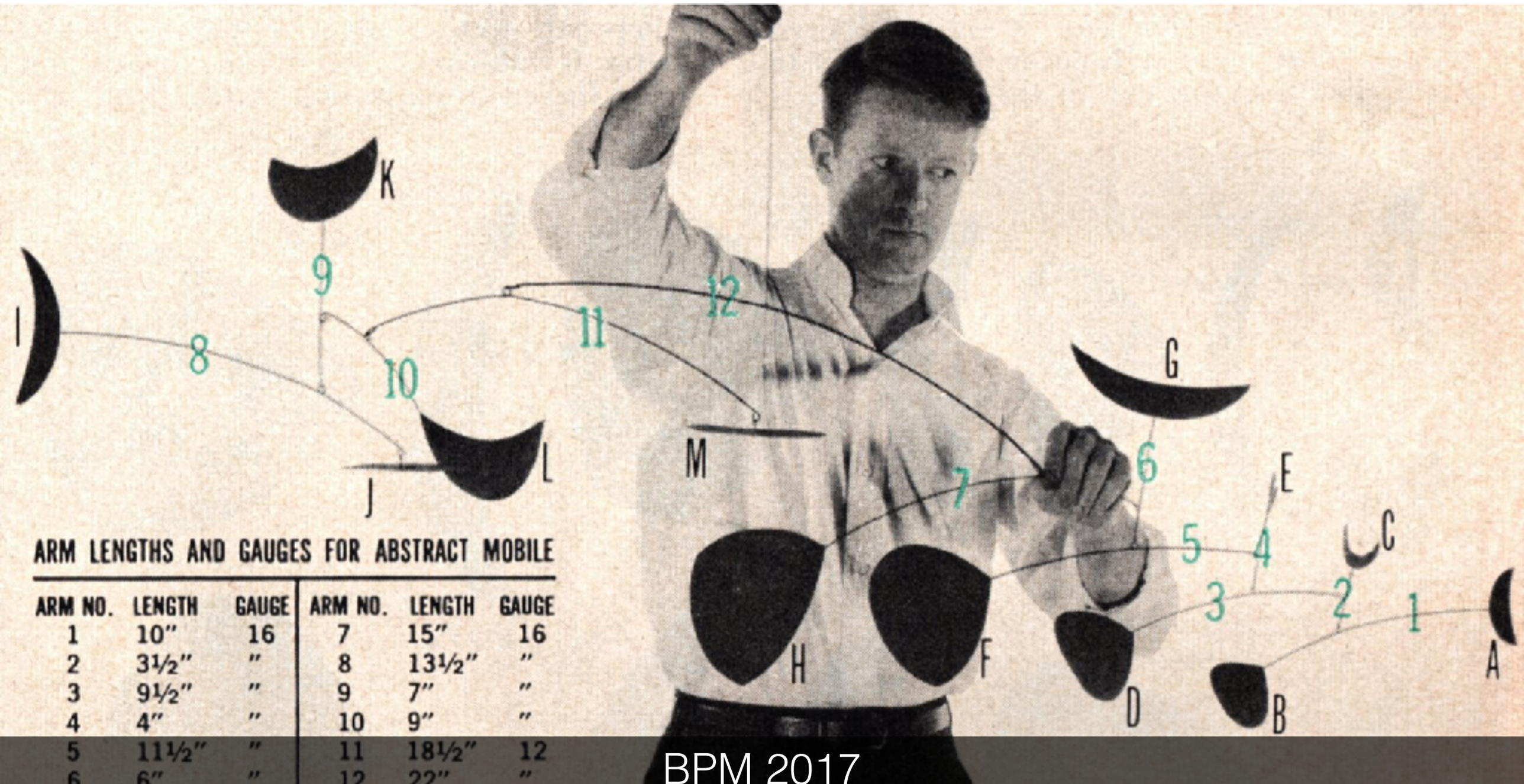
unibz

Diego Calvanese Marco Montali

{calvanese,montali}@inf.unibz.it

Free University of Bozen-Bolzano

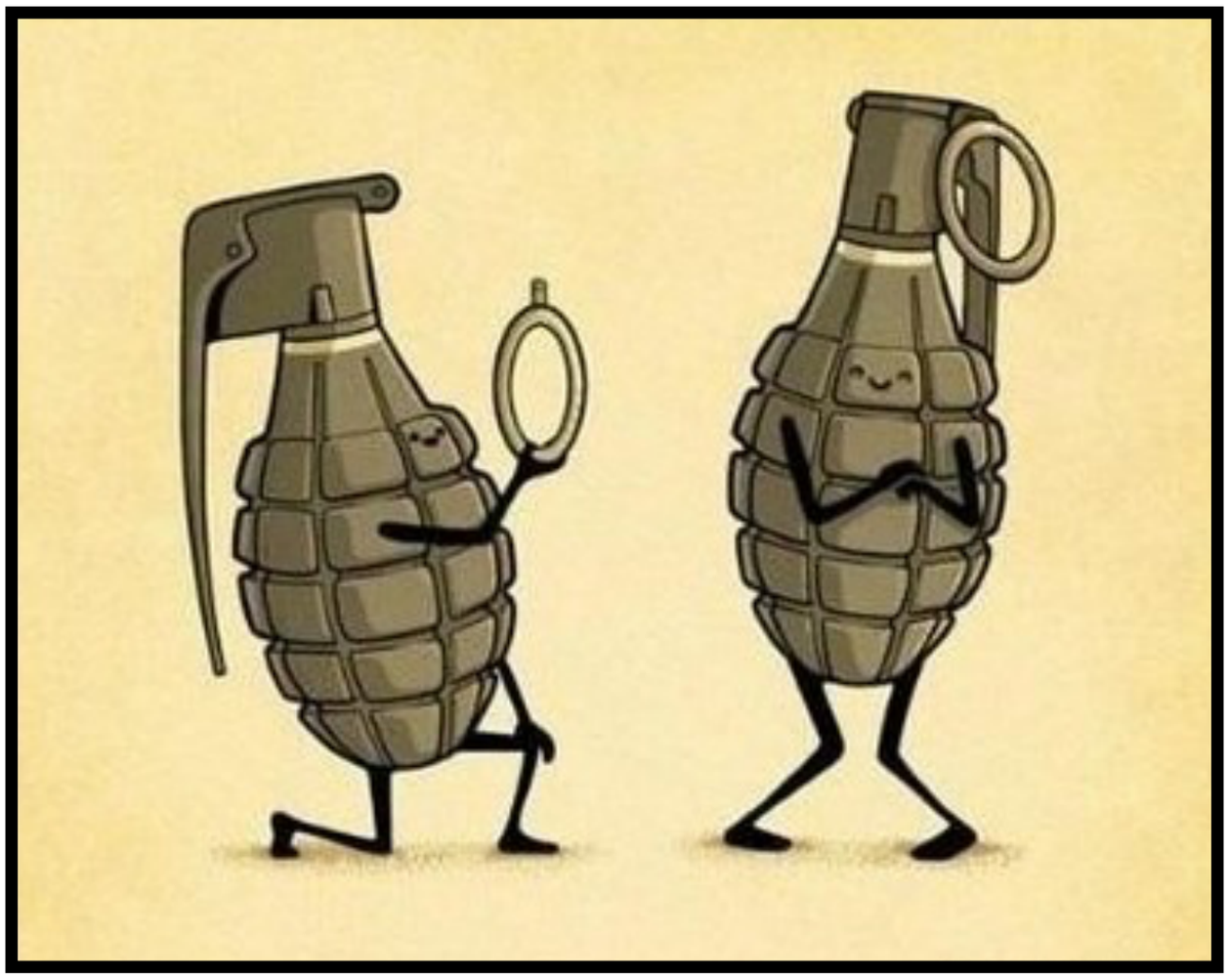
KRDB



ARM LENGTHS AND GAUGES FOR ABSTRACT MOBILE

ARM NO.	LENGTH	GAUGE	ARM NO.	LENGTH	GAUGE
1	10"	16	7	15"	16
2	3½"	"	8	13½"	"
3	9½"	"	9	7"	"
4	4"	"	10	9"	"
5	11½"	"	11	18½"	12
6	6"	"	12	22"	"

Marrying **processes** and **data**
is extremely **challenging**….



… but is a **must**
if we want to really **understand**
how **complex dynamic systems** operate.

Two Questions

How to **formally** and **conceptually**
account for the **process+data** interplay?

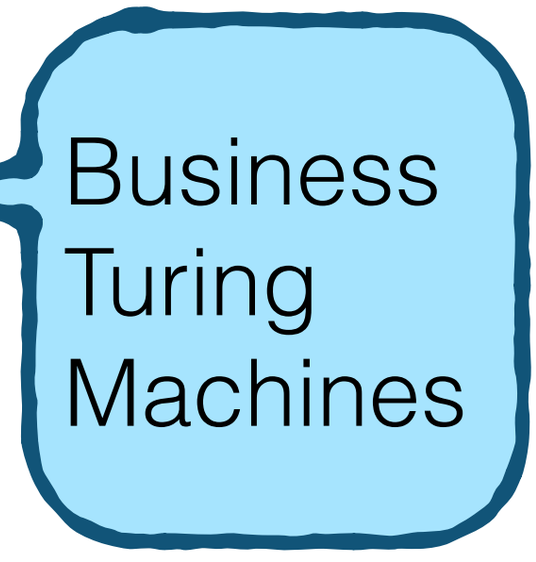
How to **verify** such **BPMs**?

N.B.: modeling and verification go side-by-side

Two Questions

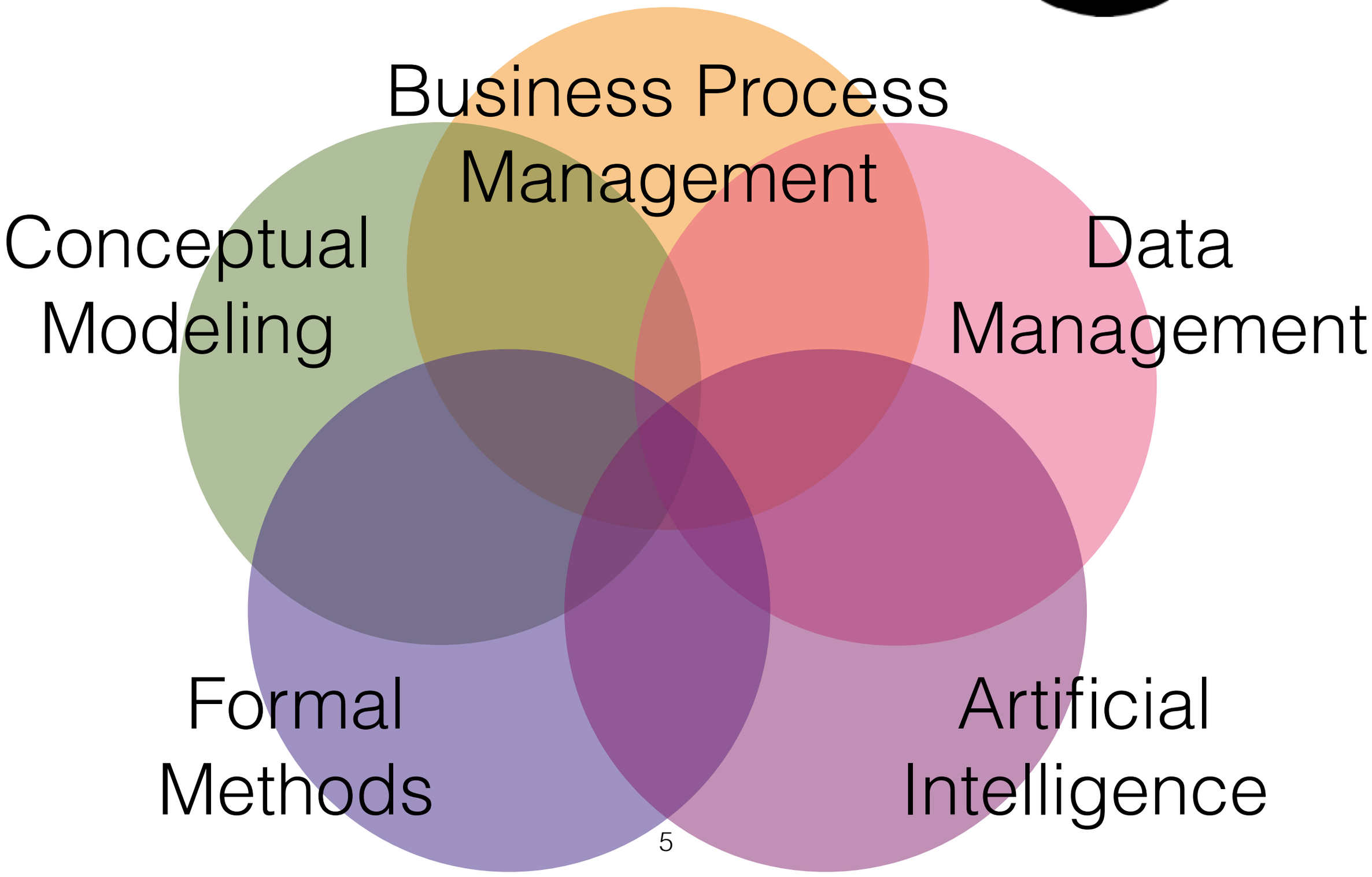
How to **formally** and **conceptually**
account for the **process+data** interplay?

How to **verify** such **BTMs** ~~**BPMs**~~?



N.B.: modeling and verification go side-by-side

Our Research at KRDB



Outline

Part 1

- Introduction and motivation: **why processes + data**
- A quick tour through the **literature** and integrated models

Part 2

- The framework of **Data-Centric Dynamic Systems**
- **Verification** results

Part 3

- **Connection** to concrete integrated models and systems
- **Concluding** remarks

Information Assets

- **Data**: the main information source about the history of the domain of interest and the relevant aspects of the current state of affairs
- **Processes**: how work is orchestrated in the domain of interest, so as to create value
- **Resources**: humans and devices responsible for the execution of work units within a process

diagnose/
get reqs.

adjust

reality

data

models

enact/
monitor

(re)
design

(knowledge)
workers

managers/
analysts

IT support

configure/
deploy

Is this Synergy Reflected by BP Methods and Models?

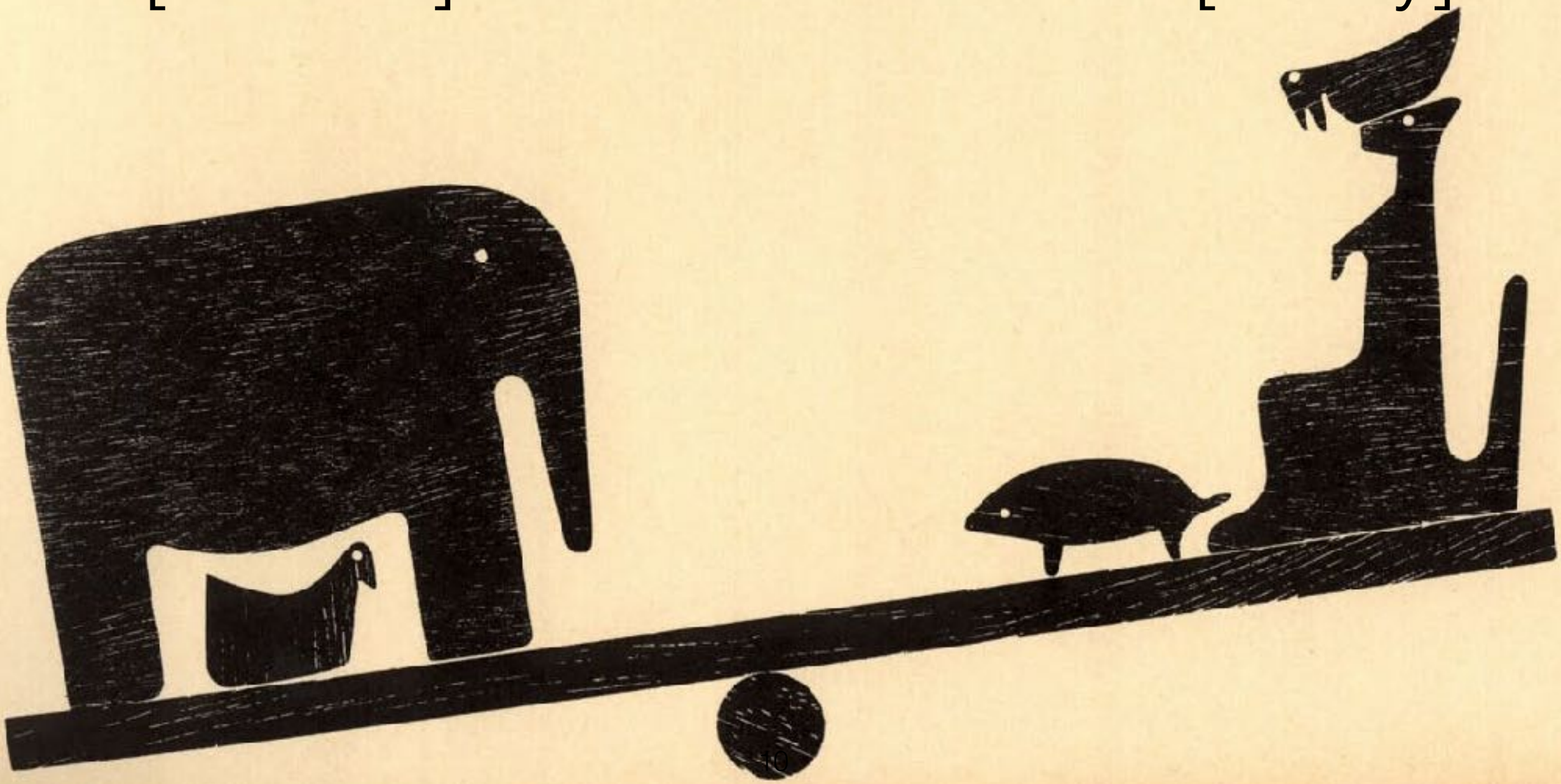
Survey by *Forrester* [Karel et al, 2009]: **lack of interaction between data and process experts**.

- *BPM professionals*: **data are subsidiary to processes**
- *Master data managers*: **data are the main driver** for the company's existence
- 83/100 companies: **no interaction at all** between these two groups
- This isolation propagates to models, languages and tools

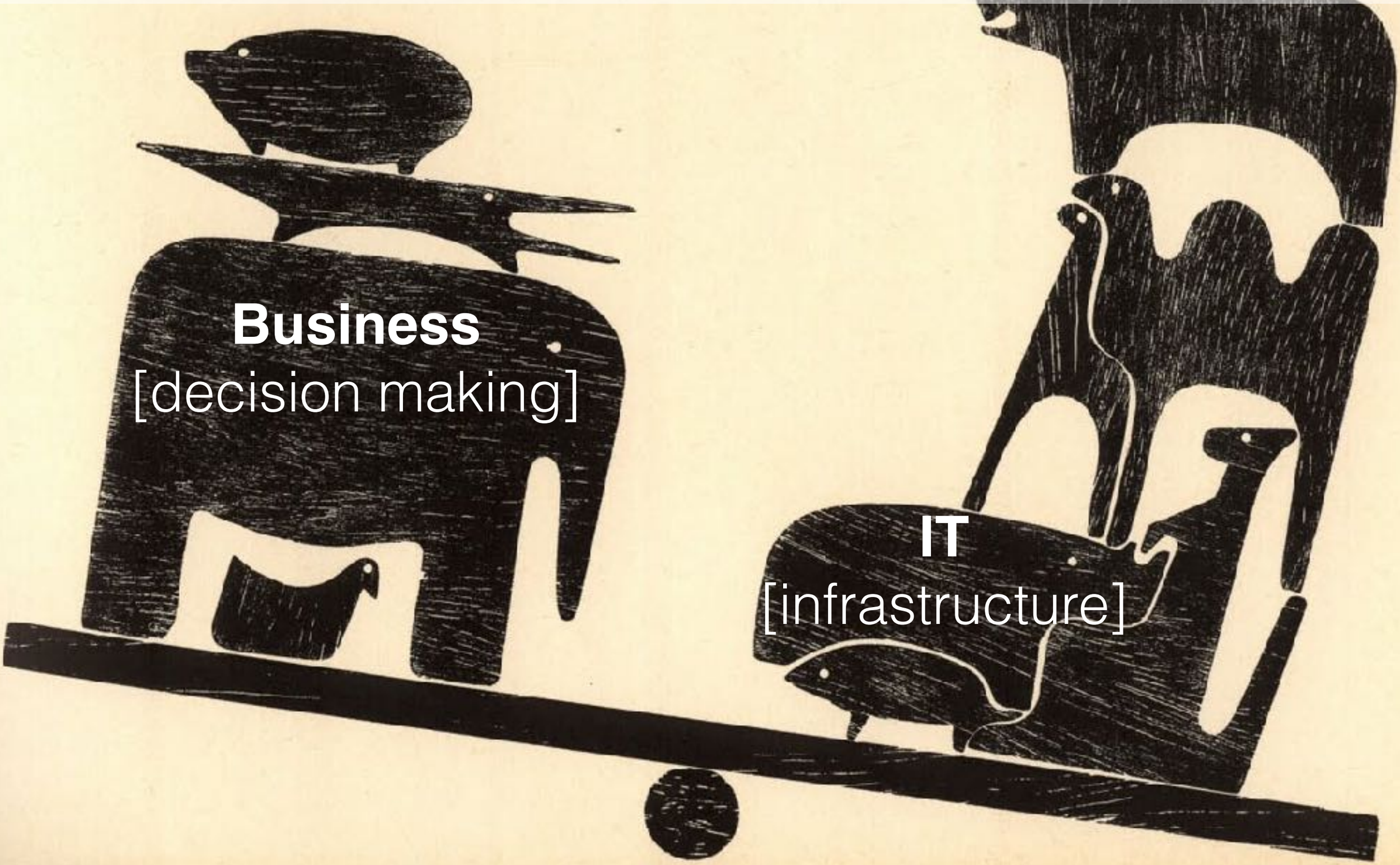
Experience Dichotomy

Management
[models]

Workers
[reality]



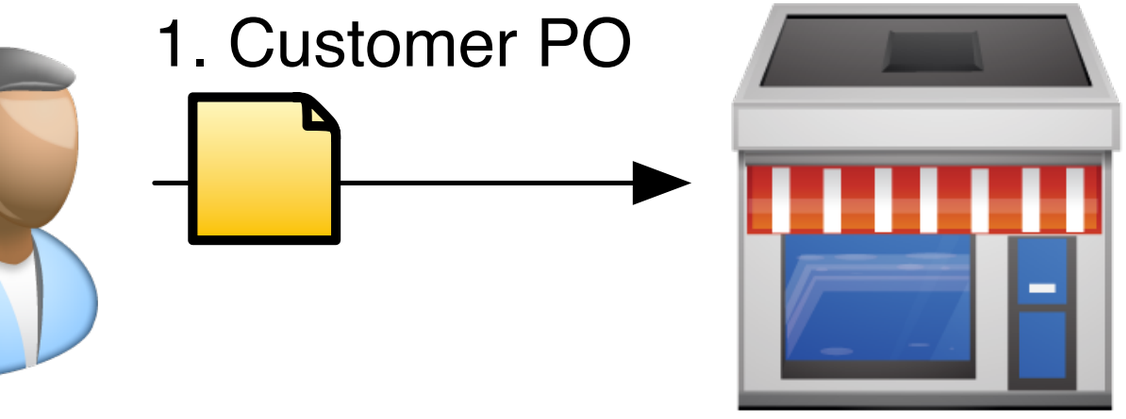
Management Dichotomy



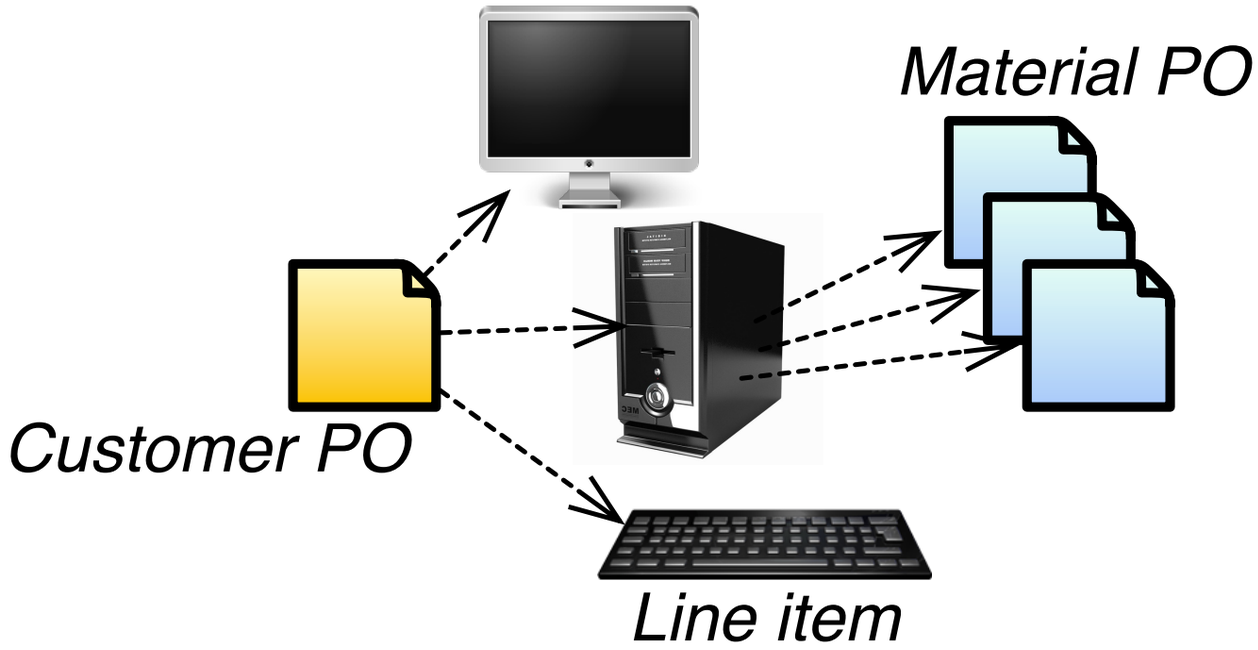
Expertise Dichotomy

A Successful Organization

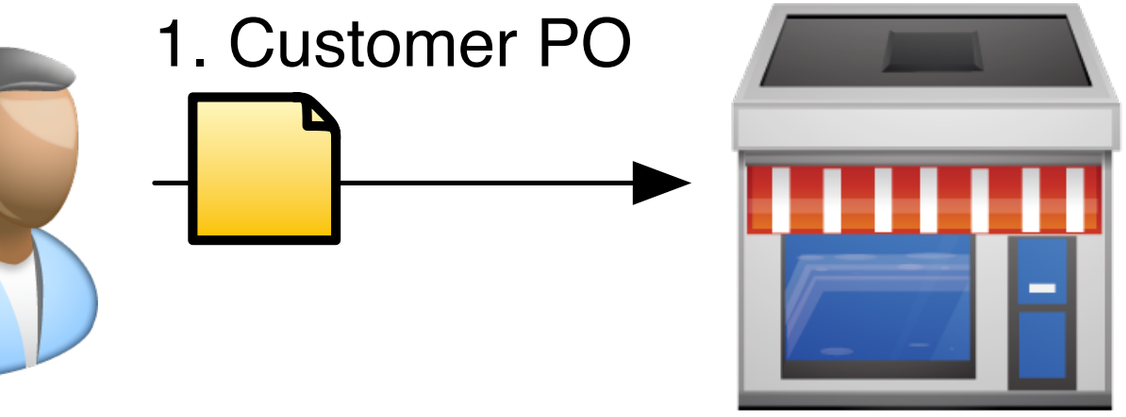
Example: Order-To-Delivery



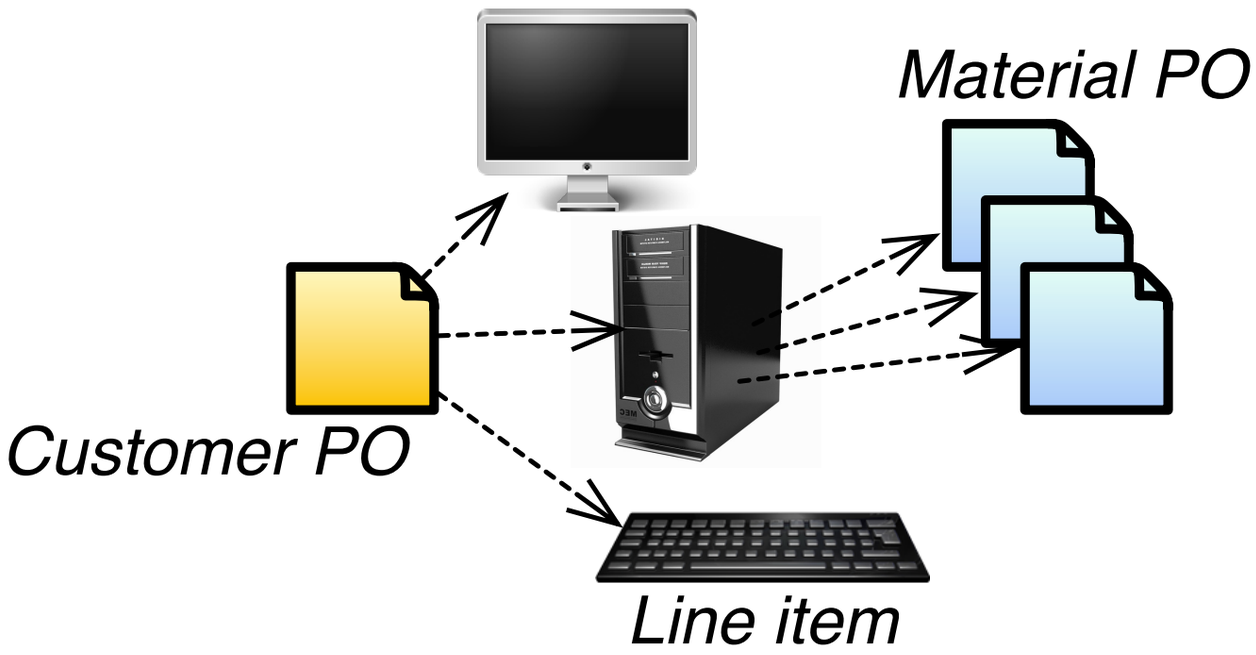
2. order decomposition



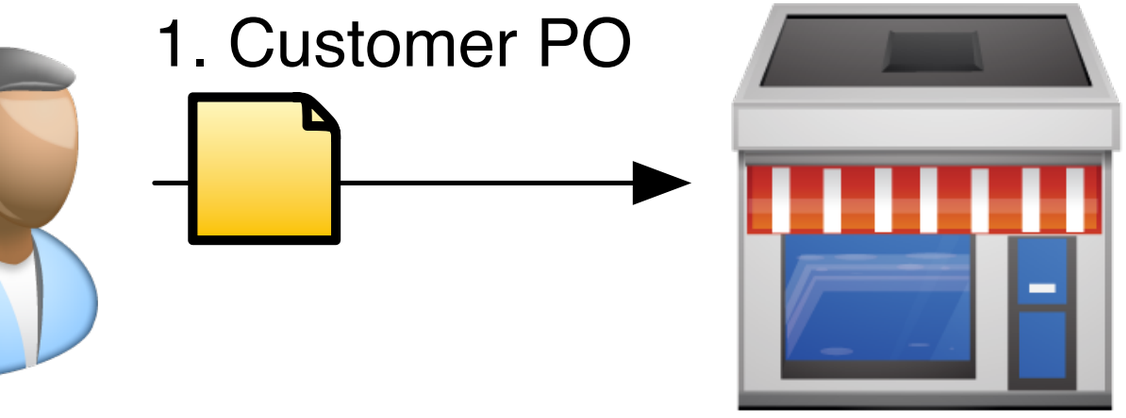
1. Customer PO



2. Order decomposition



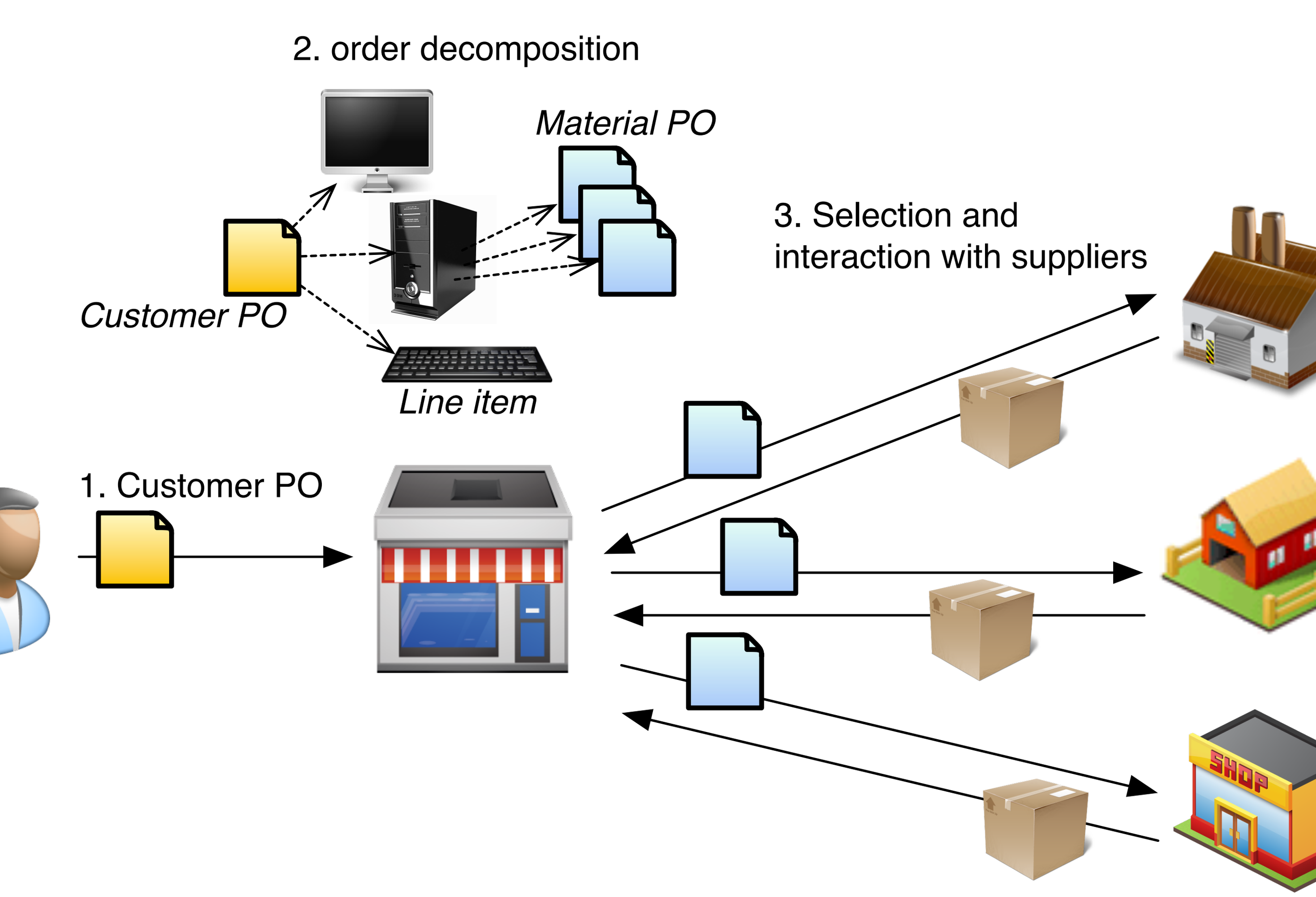
1. Customer PO



3. Selection and interaction with suppliers

2. order decomposition

3. Selection and interaction with suppliers



1. Customer PO

2. order decomposition

Material PO

Customer PO

Line item

3. Selection and
interaction with suppliers

1. Customer PO

SHOP

4. material assembly

2. order decomposition

Material PO

Customer PO

Line item

3. Selection and interaction with suppliers

1. Customer PO

5. Shipment

4. material assembly

Observations

- A complex process, where the company acts as an intermediate hub between customers and suppliers

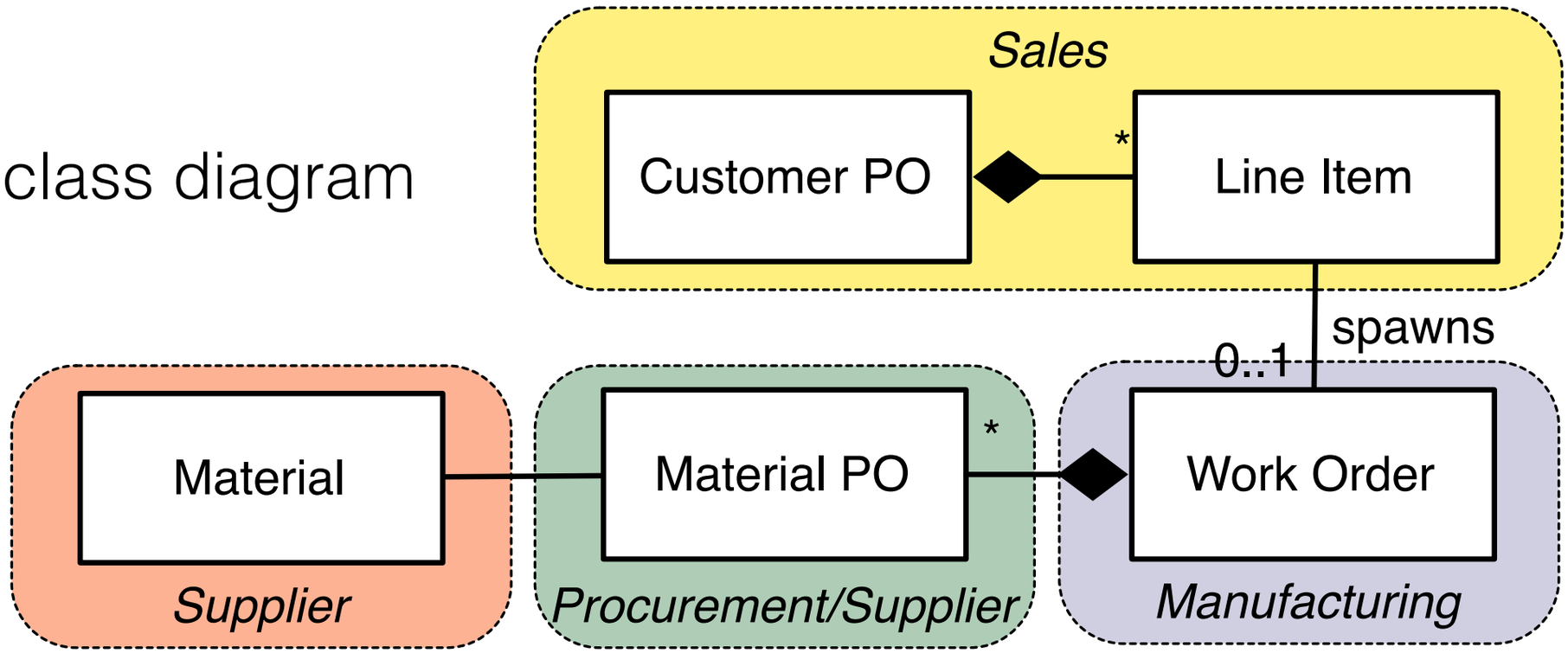
- **Happy path**
  1) The customer issues a purchase order
  2) The ordered material is obtained from suppliers
  3) The material is shipped, possibly using different packages

- One **exceptional path** (in general, there are many):
  1) The customer cancels the order
  2) A **cancelation policy** is applied to calculate a penalty

Conventional Data Modeling

Focus: revelant entities, relations, *static* constraints

UML class diagram

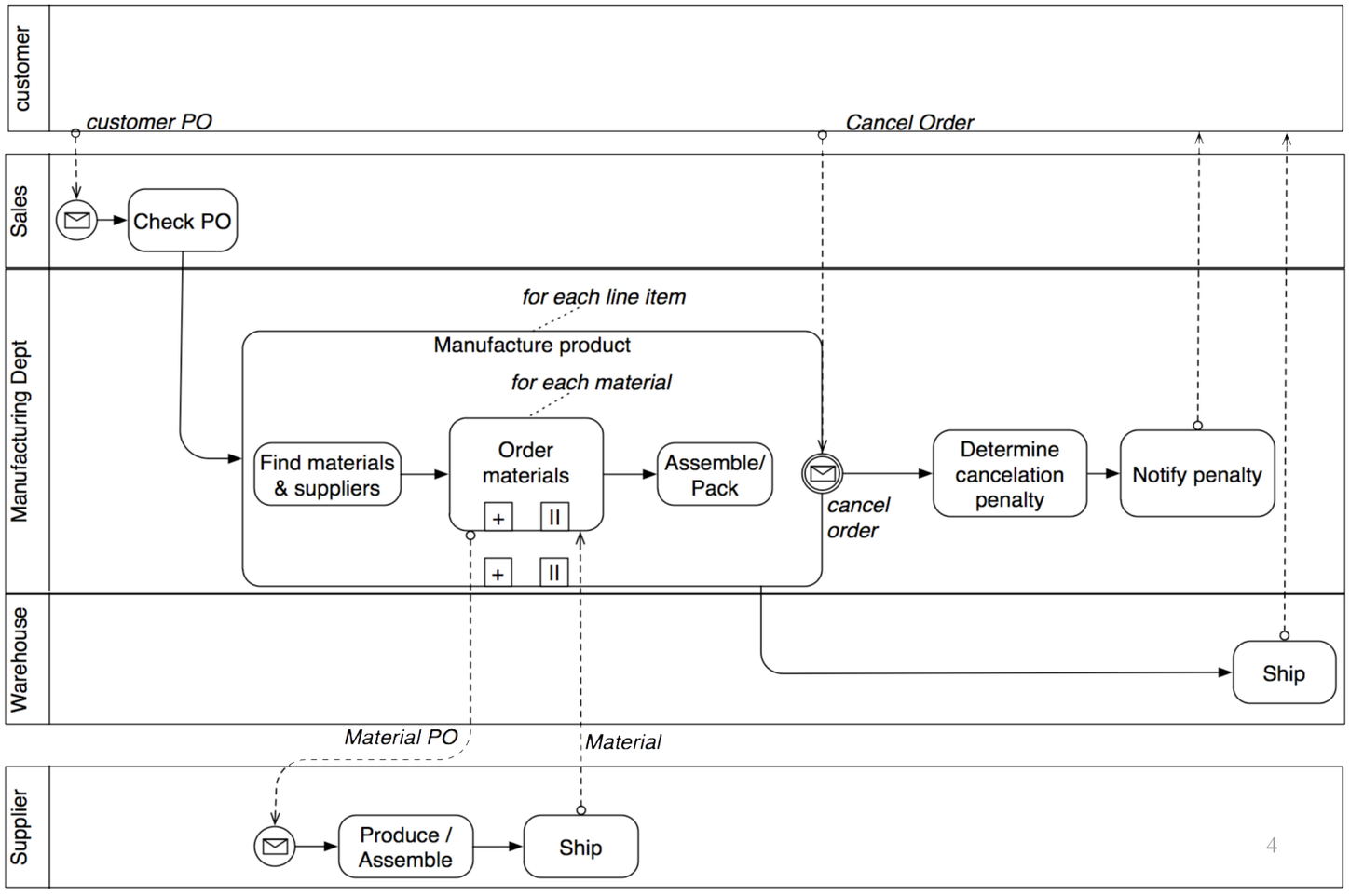


But… how do data evolve?
Where can we find the “state” of a purchase order?

Conventional Process Modeling

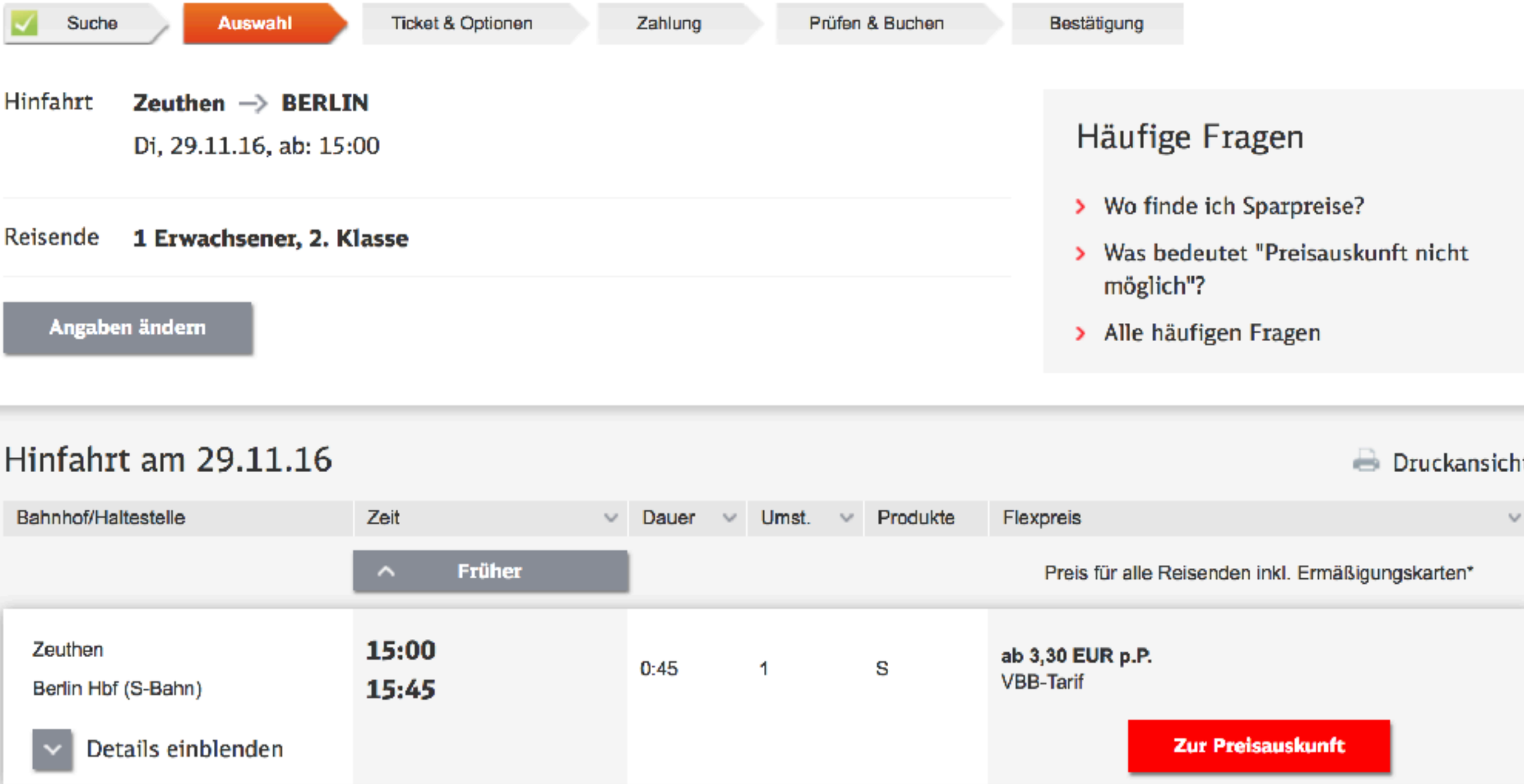
Focus: control-flow of activities in response to events

BPMN
collaborative
process

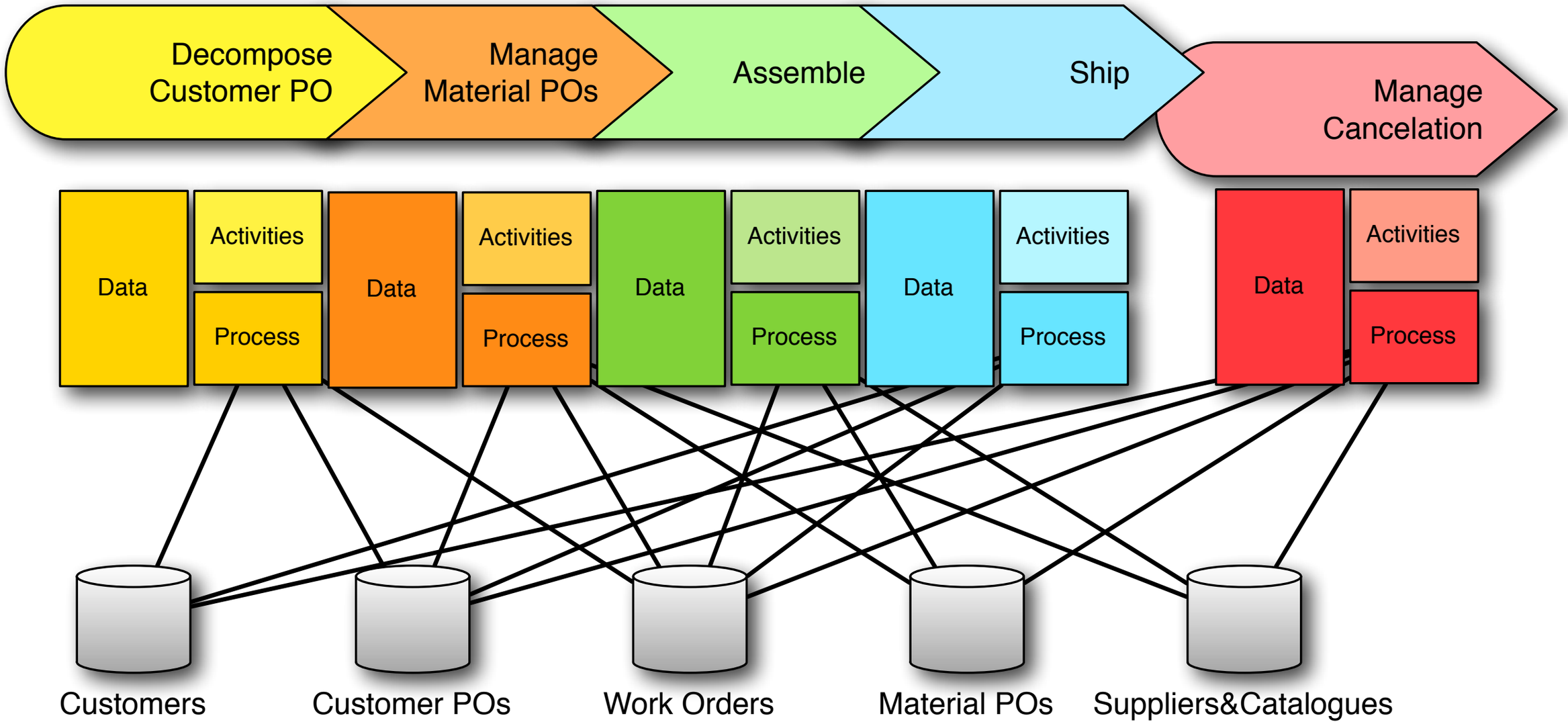


But… how do activities update data?
What is the impact of canceling an order?

A Deployed Process



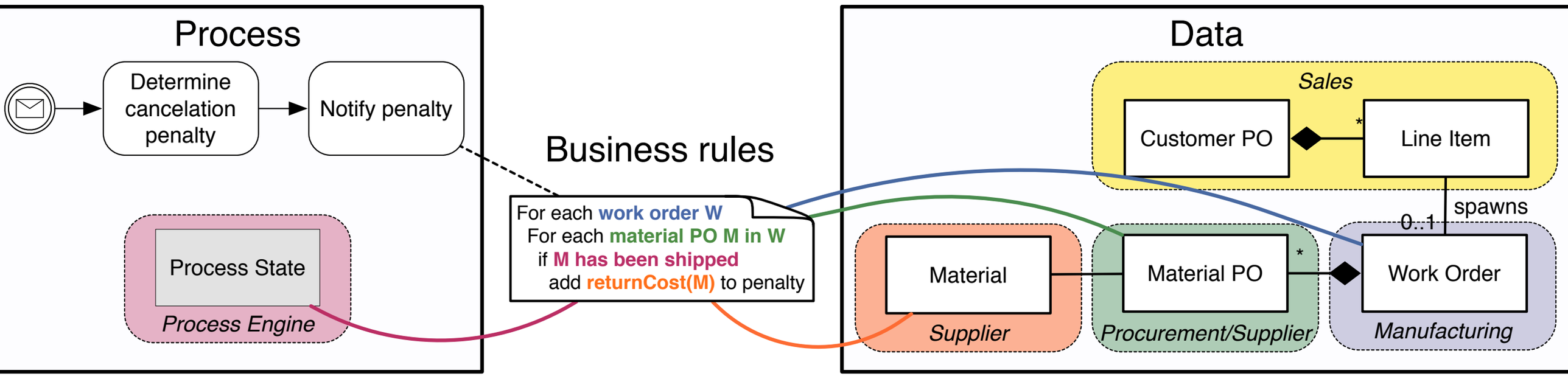
Do you like Spaghetti?



IT integration: difficult to manage, understand, maintain

Too Late!

- Where are the data?
- Where shall we model relevant business rules?
- Consider an **order cancelation policy** that needs to check which material has been already shipped towards determining the customer penalty…

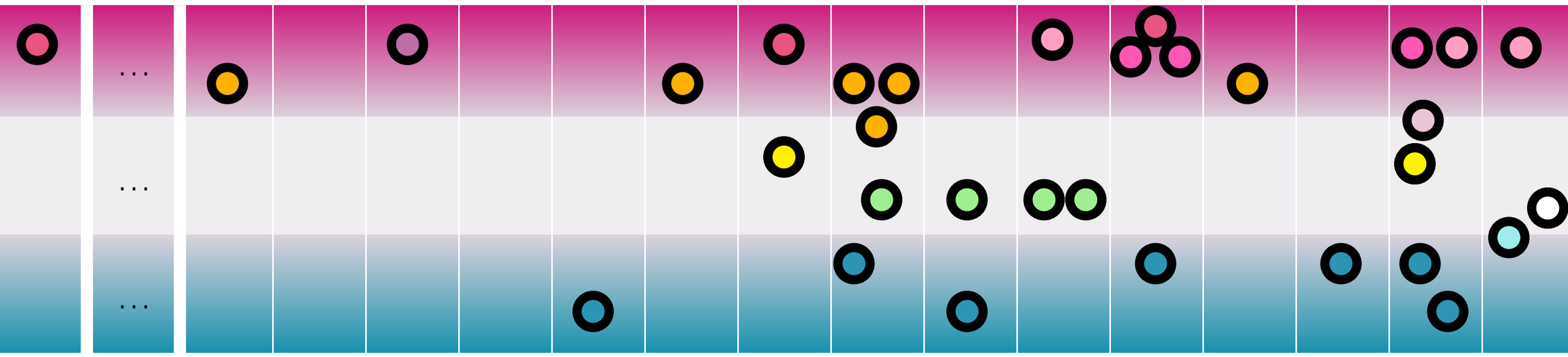


A LONG TIME AGO IN A GALAXY FAR FAR AWAY...

N.B.: these are “sparse” dots!!!

…There is Hope!

data-centric

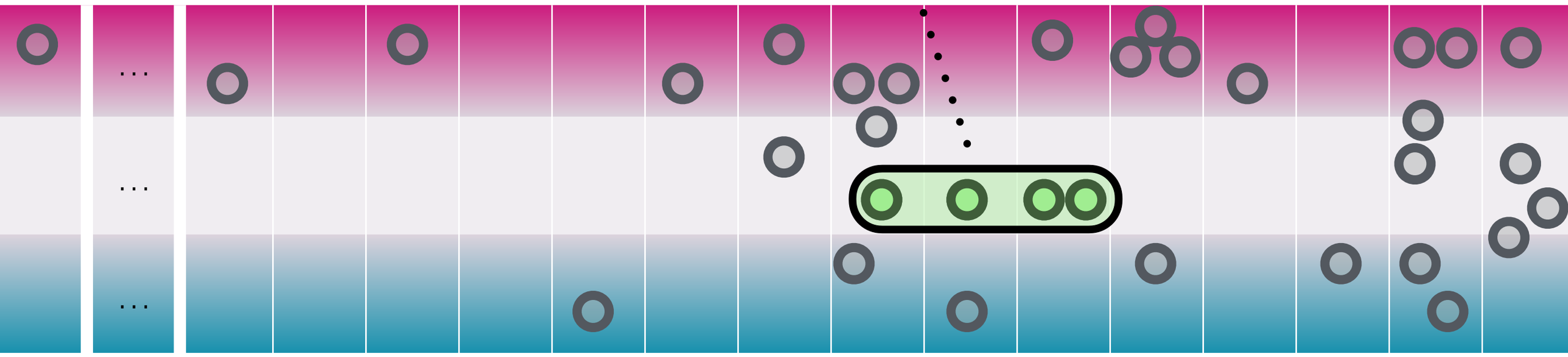


activity-centric

1		2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
9		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	…	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
8		3	4	5	6	7	8	9	0	1	2	3	4	5	6	7

- [BPM2010, Richardson]: **BPM vs master data dichotomy**
- **Data+Process integration** key to:
  - assess **value of processes** and **evaluate KPIs** [Meyer et al, 2011]
  - **aggregate** relevant **info**, elicit **business rules** [ABDIS11, Dumas]
- [Reichert, 2012]: **"Process and data are just two sides of the same coin"**

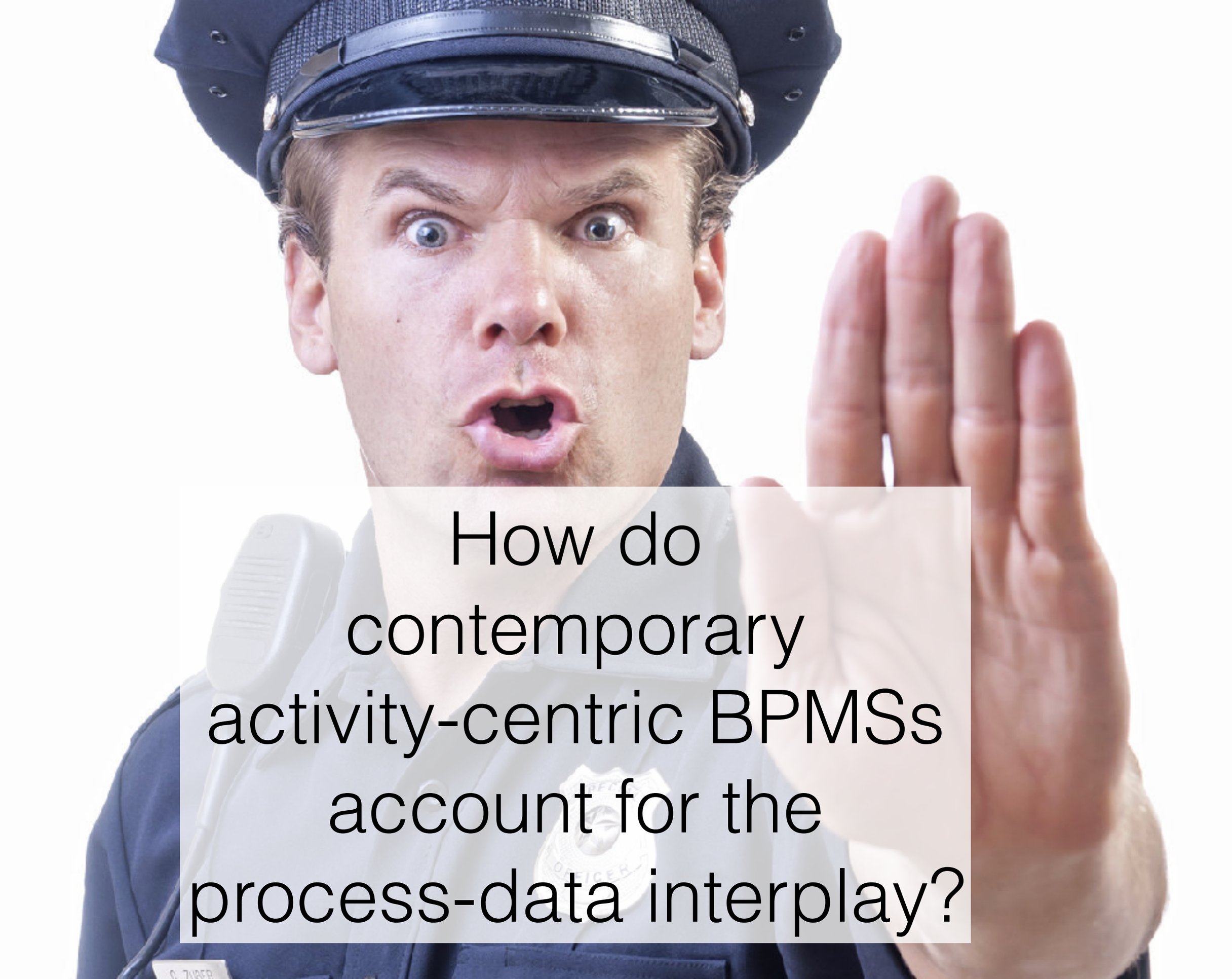
data-centric ⋮



activity-centric

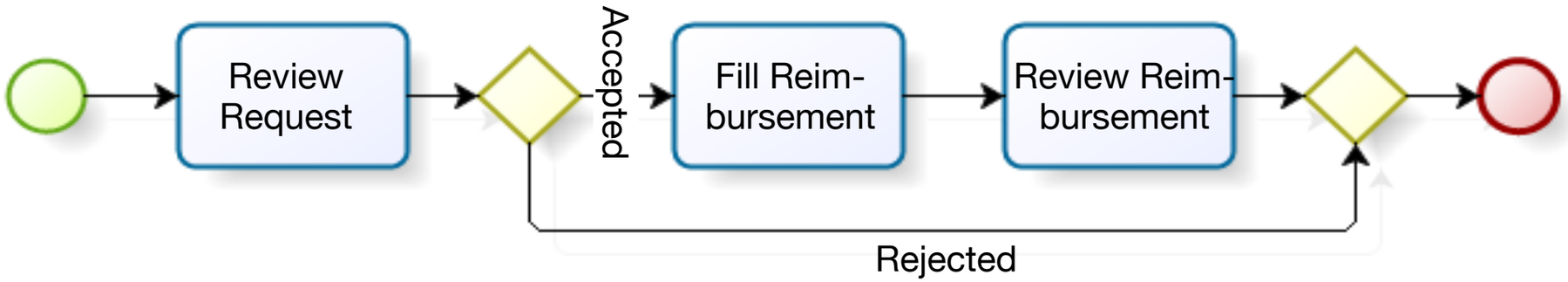
1		2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
9		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	...	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
8		3	4	5	6	7	8	9	0	1	2	3	4	5	6	7

Before moving to exotic models…

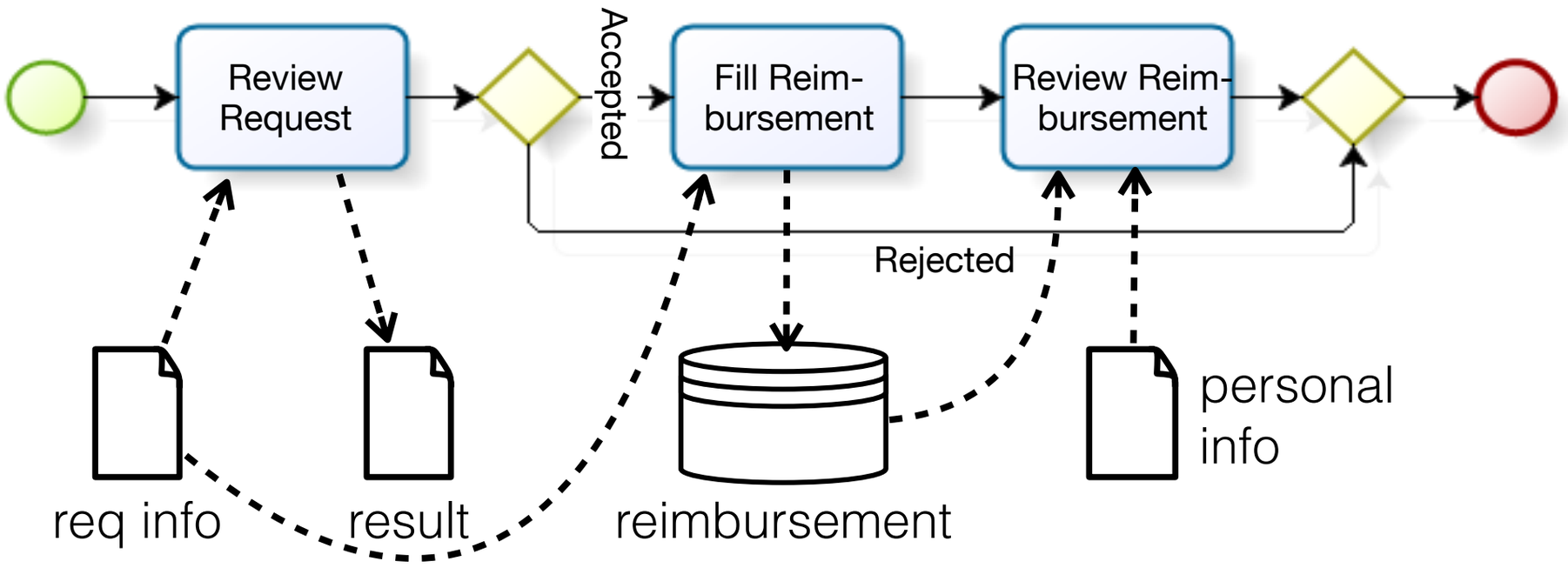


How do contemporary activity-centric BPMSs account for the process-data interplay?

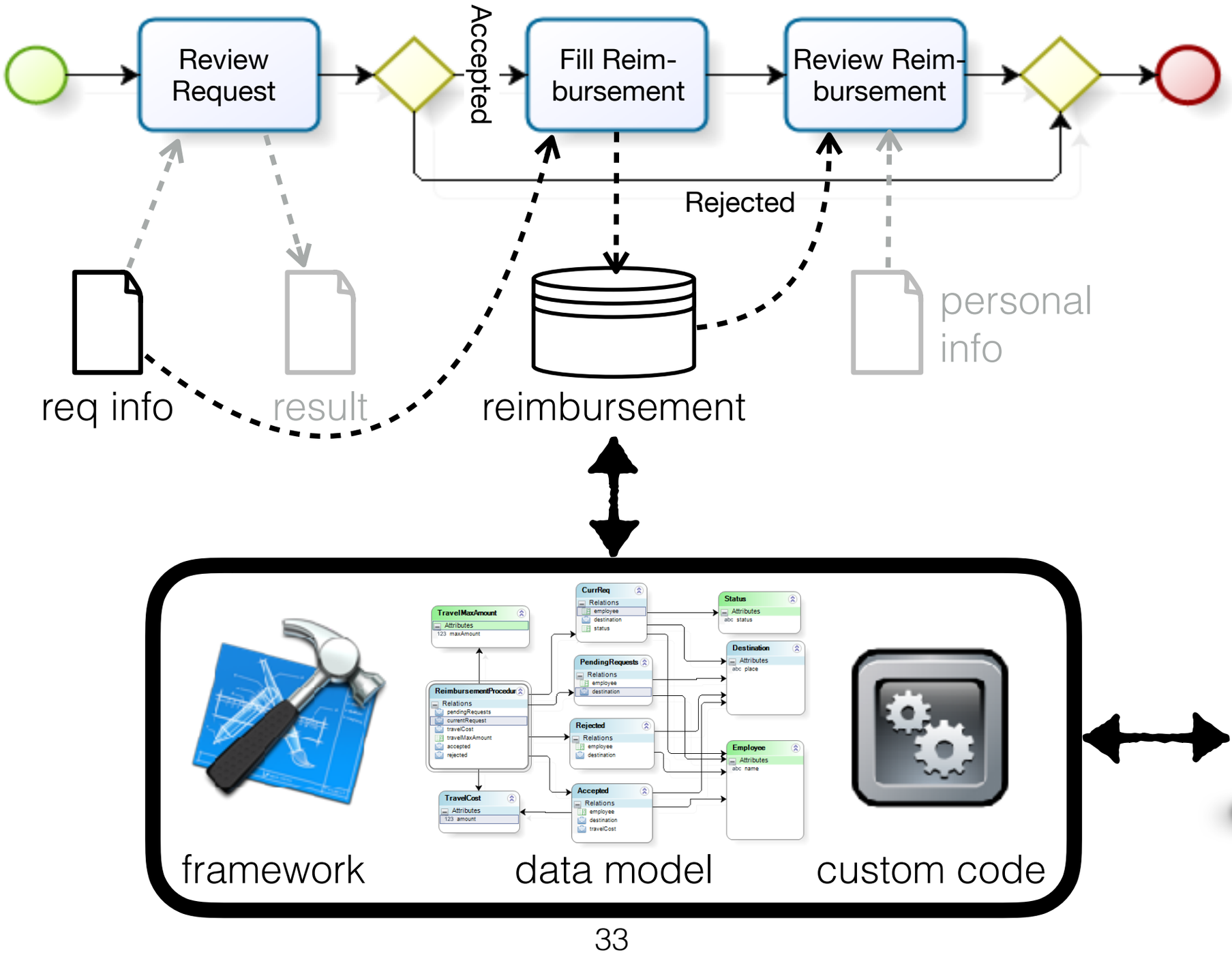
Example: BizAgi (~)



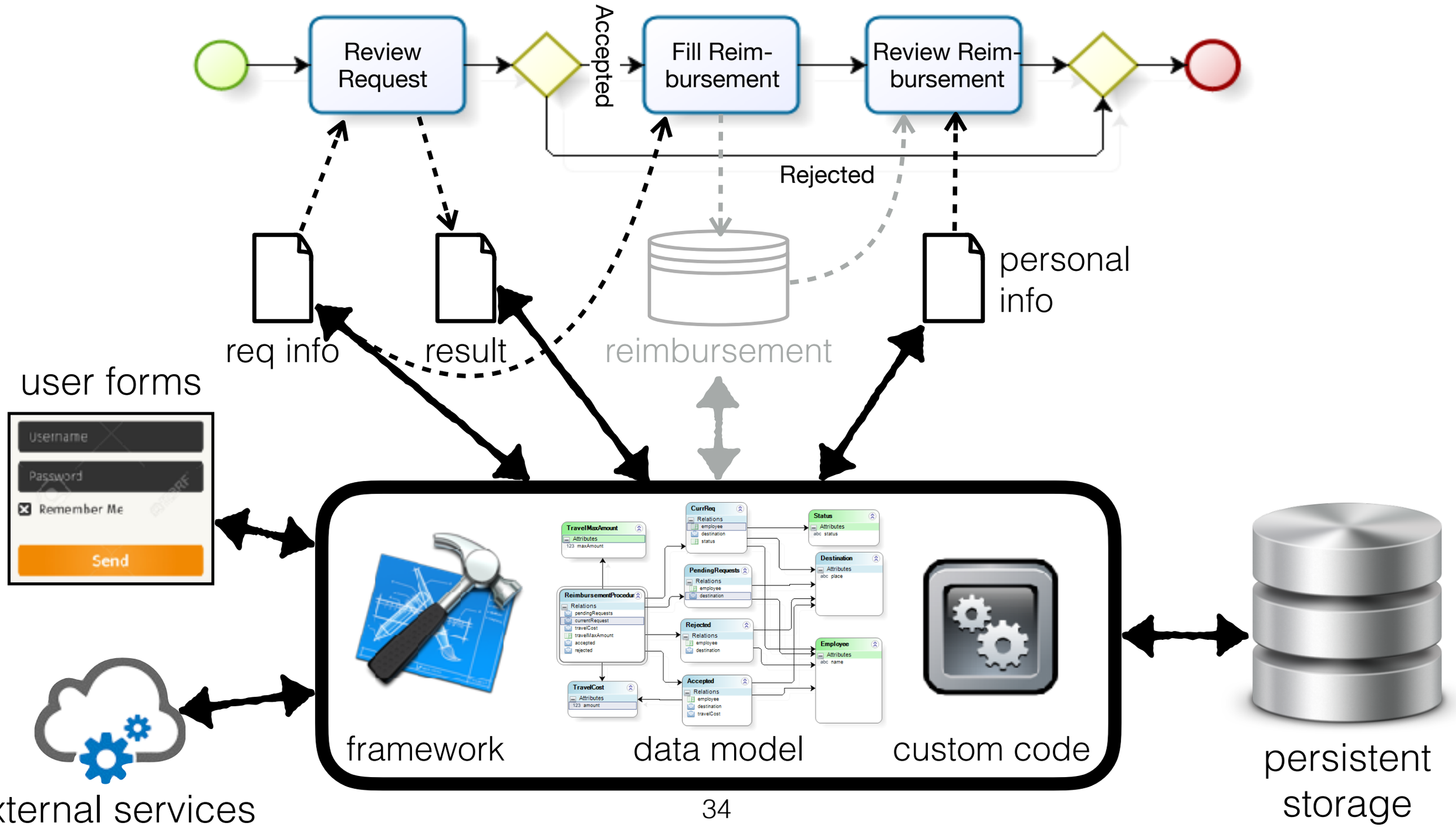
Case and Persistent Data



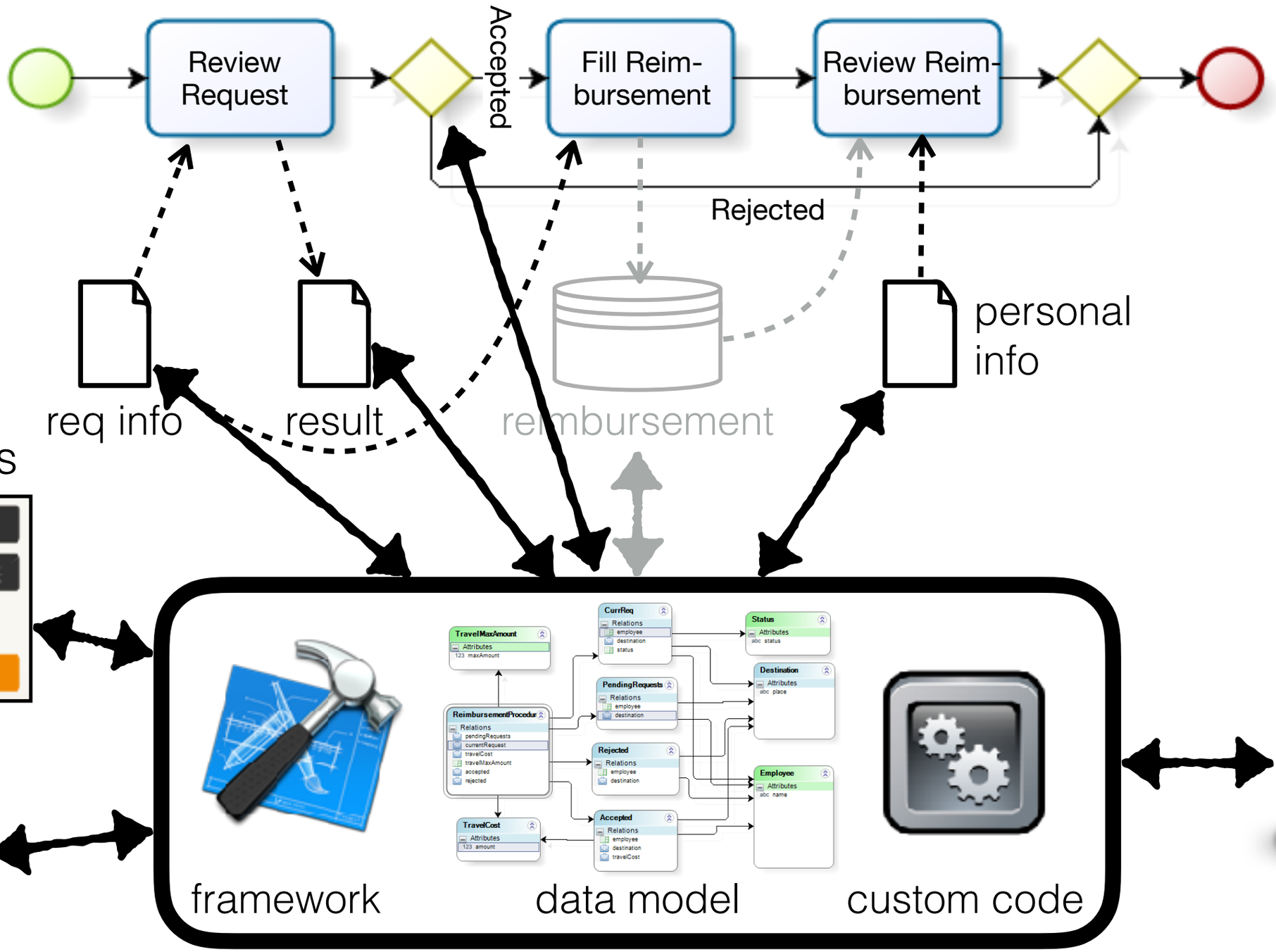
Persistent Data Engineering



Case Data Engineering



Decision Modeling



A General Recipe

“REAL” PROCESS

- Explicit control-flow
- Local, case data
- Global, persistent data
- Queries/updates on the persistent data
- External inputs
- Internal generation of fresh IDs

Cooking with Standard Process Languages

BPMN

- ✓ Explicit control-flow
- ~ Local, case data
- ~ Global, persistent data
- ✗ Queries/updates on the persistent data
- ✗ External inputs
- ✗ Internal generation of fresh IDs

Business Process

A set of logically related **tasks** performed to achieve a defined business outcome for a particular customer or market.

(Davenport, 1992)

A collection of **activities** that take one or more kinds of input and create an output that is of value to the customer.

(Hammer & Champy, 1993)

A set of **activities** performed in coordination in an organizational and technical environment. These activities jointly realize a business goal.

(Weske, 2011)

Business Process

A set of logically related **tasks** performed to achieve a defined business

ort, 1992)

A colle nput and create a

py, 1993)

A set of nizational and tech ze a business goal.

(Weske, 2011)

Task logic:
tightly intertwined
with data updates!

[IBM J., Nigam and Caswell] **Business Artifacts**

[BPM10WS, Estanol et al] First paper on **BAUML**

[WSFM10, Hull et al.] **First paper on IBM GSM**

Kick-off of the **EU Project ACSI**

First draft of **OMG CMMN**

[OTM08, Hull] **Survey on business artifacts**

ata-centric

[BPM09WS, Kūnzle and Reichert] First paper on **Philharmonic Flows**

[TMIS16, Sun et al] **Universal Artifacts**

activity-centric

[BPM16Forum, Hewelt and Weske] First paper on **Chimera**

[CAiSE17, De Giacomo et al] **BPMN with data**

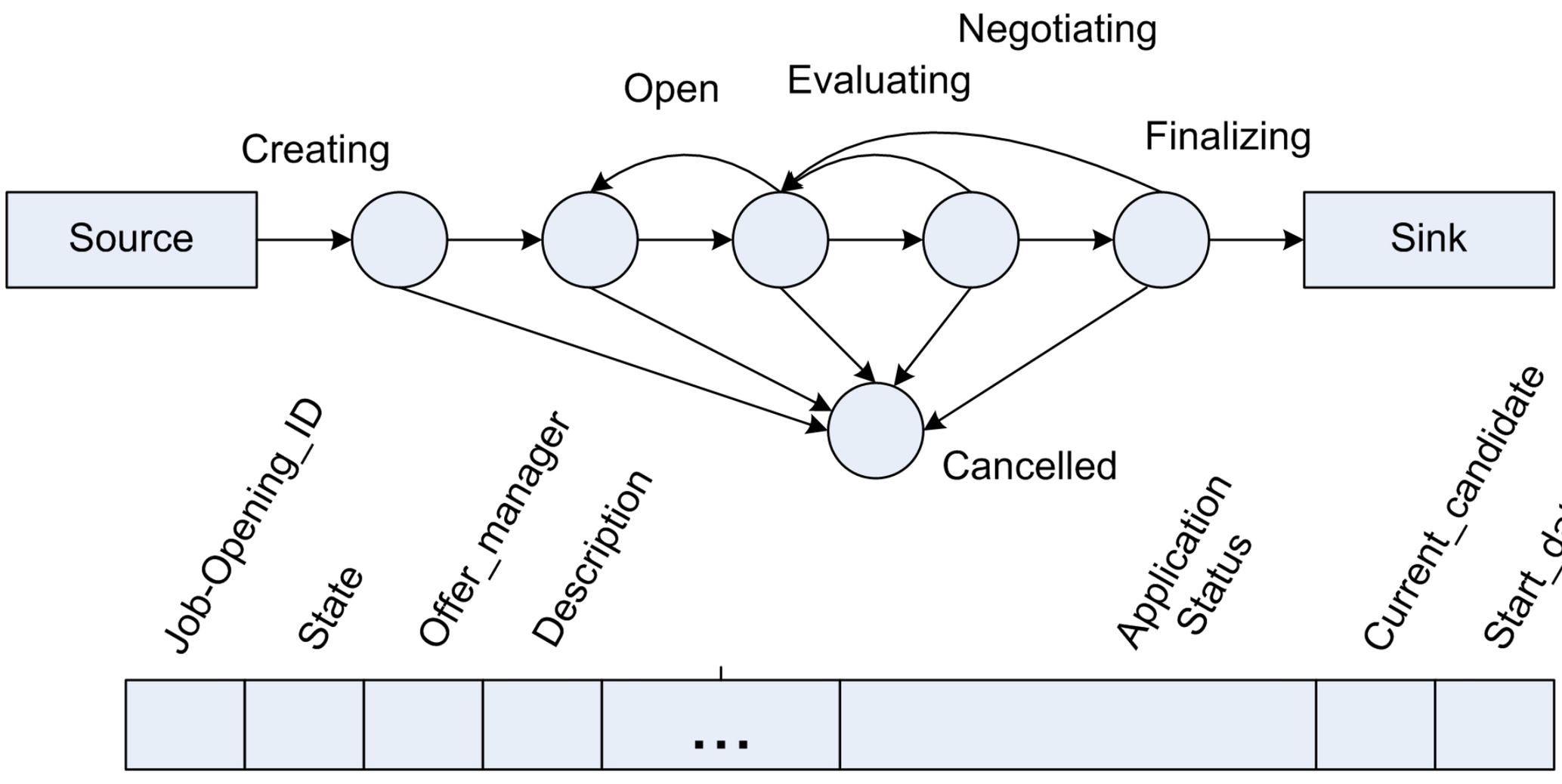
1		2	2	2	2											
9		0	0	0	0											
9	…	0	0	0	0											
8		3	4	5	6	7	8	9	0	1	2	3	4	5	6	7

Business Entities/Artifacts

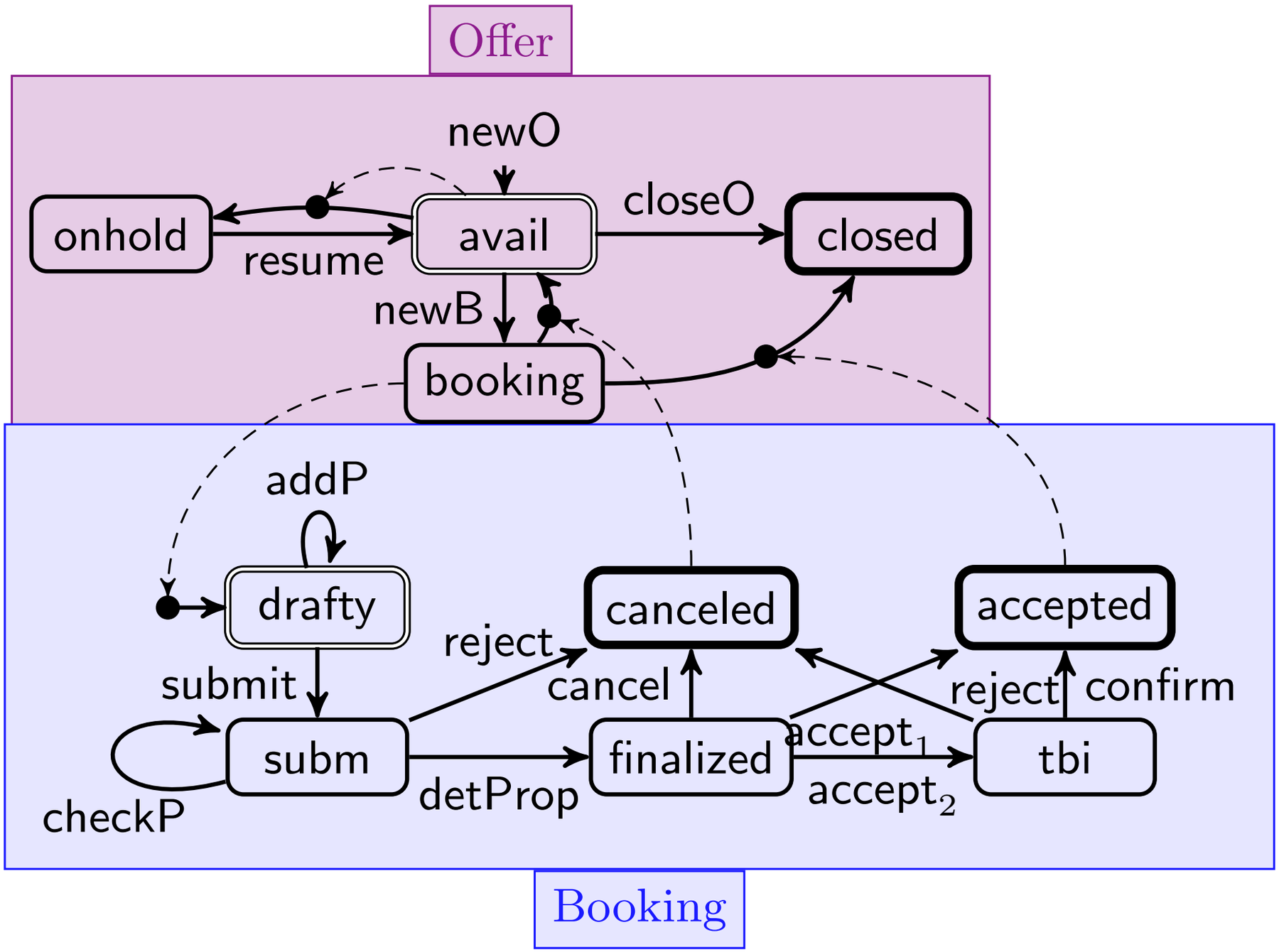
Data-centric paradigm for process modeling

- First: *elicitation of relevant business entities* that are evolved within given organizational boundaries
- Then: definition of the *lifecycle* of such entities, and how *tasks trigger the progression* within the lifecycle
- Active research area, with concrete languages (e.g., IBM GSM, OMG CMMN)
- Cf. **EU project ACSI** (completed)

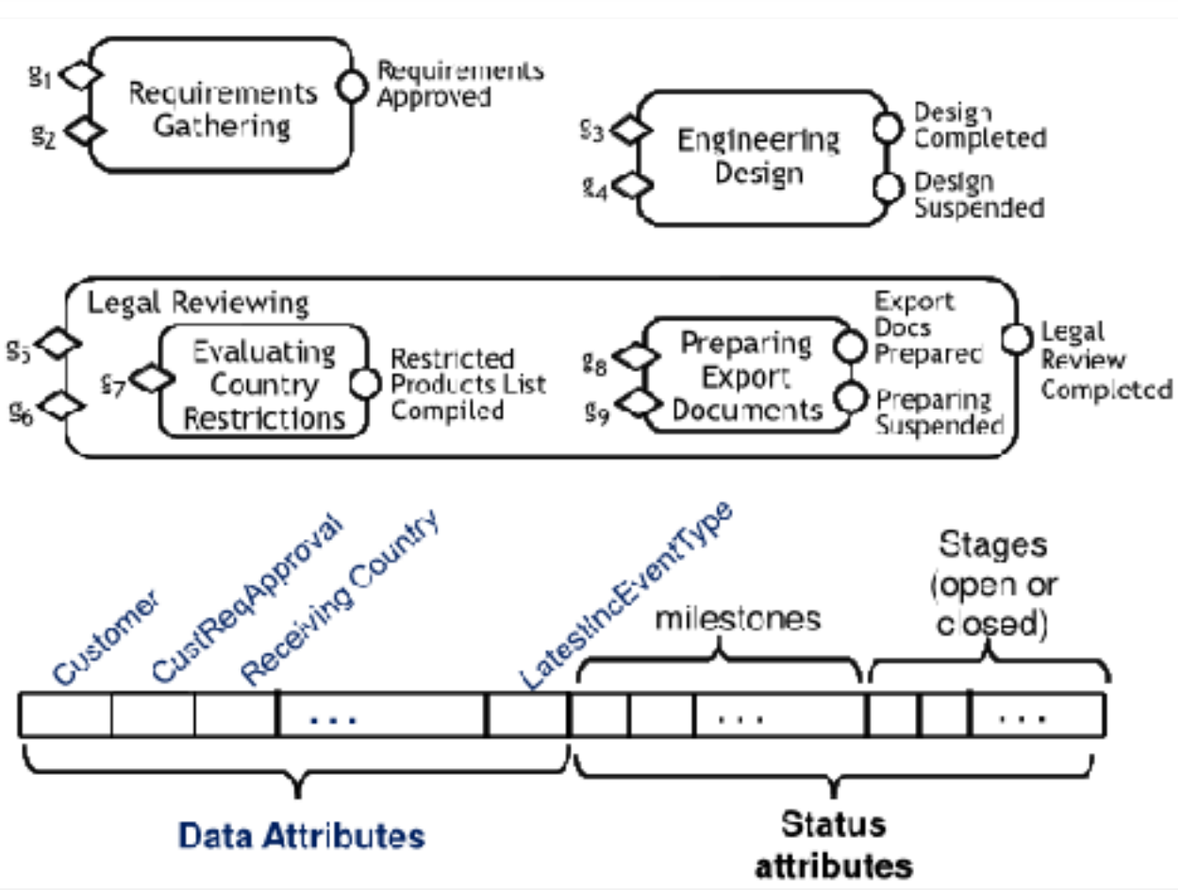
Finite-State Machines



Synchronization

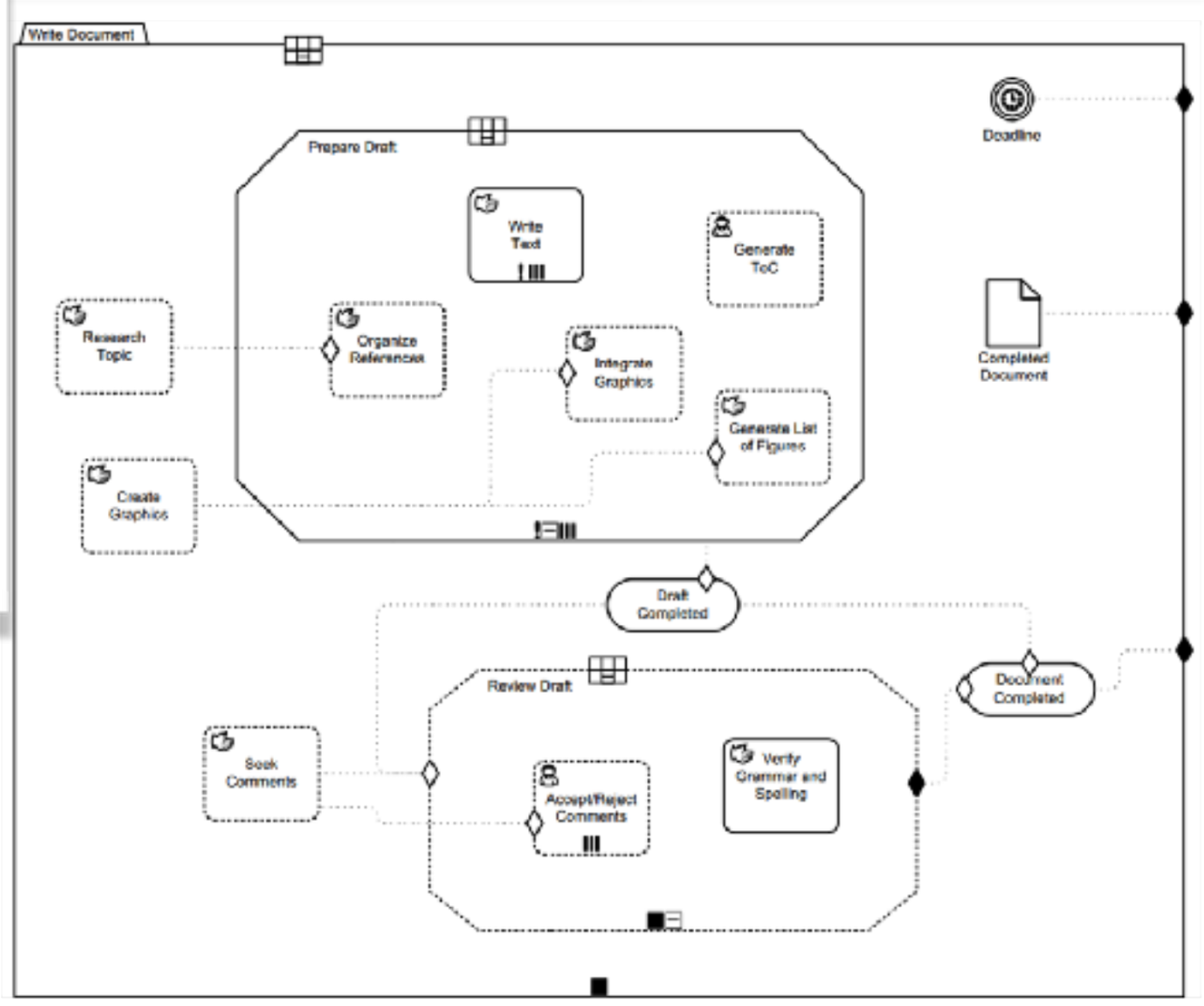


GSM - CMMN

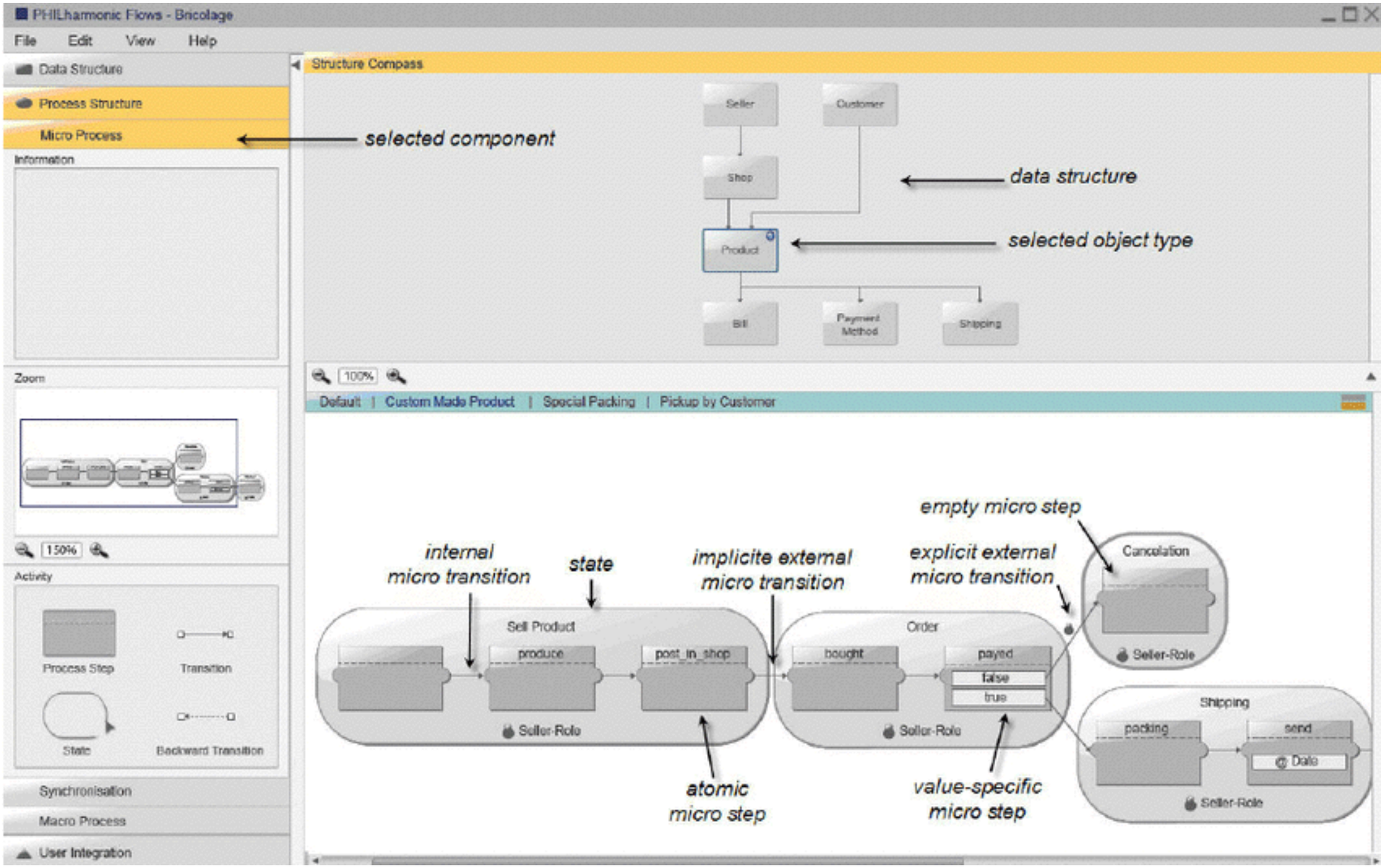


Guard Stage Milestone

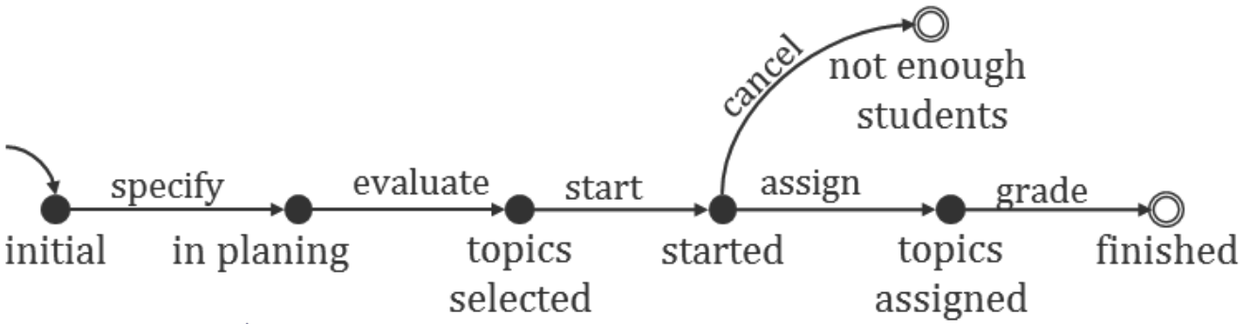
Case Management Model and Notation

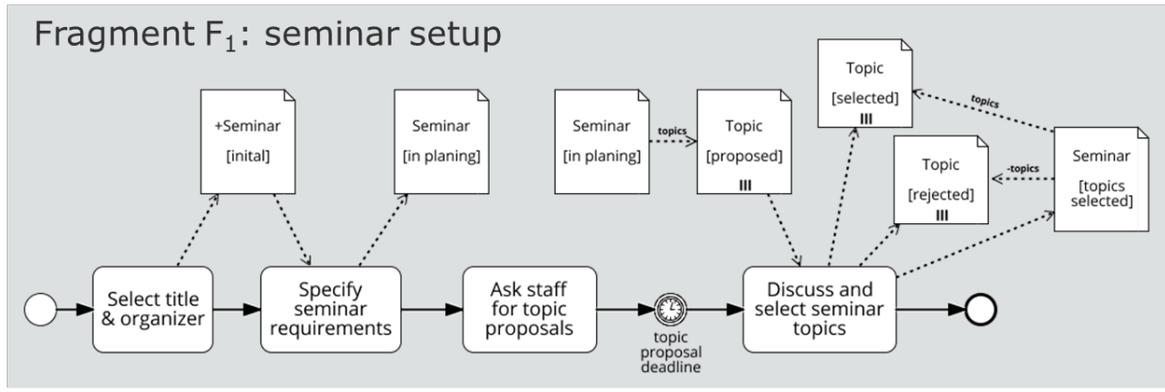


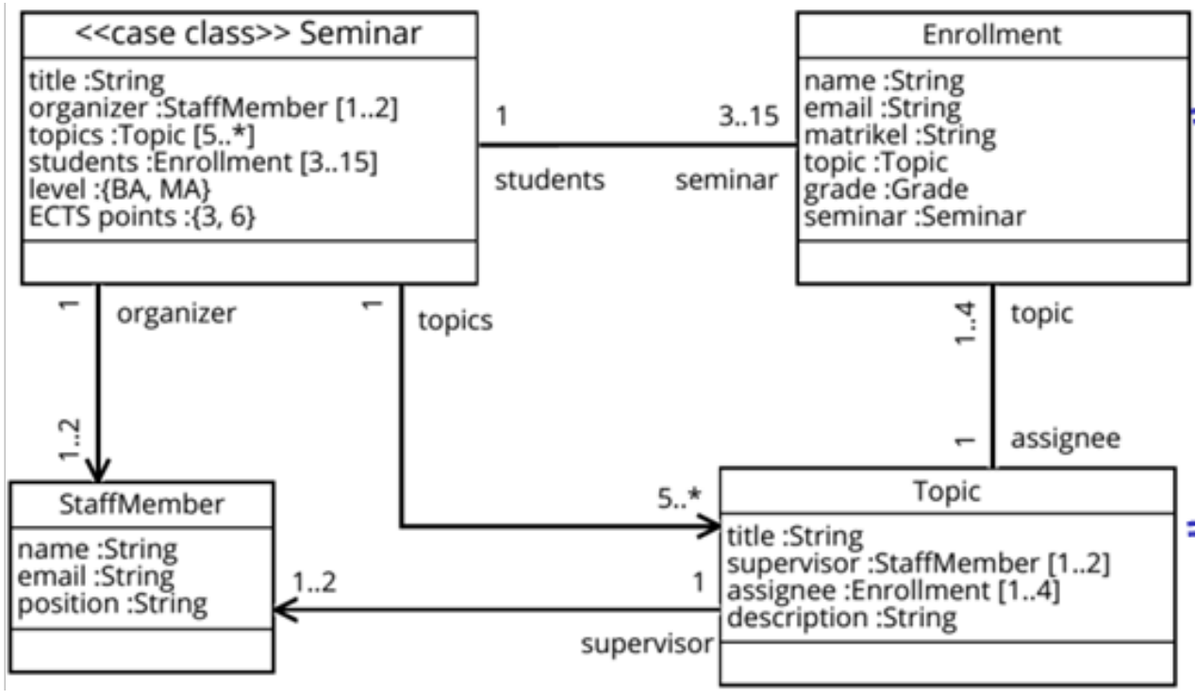
Philharmonic Flows

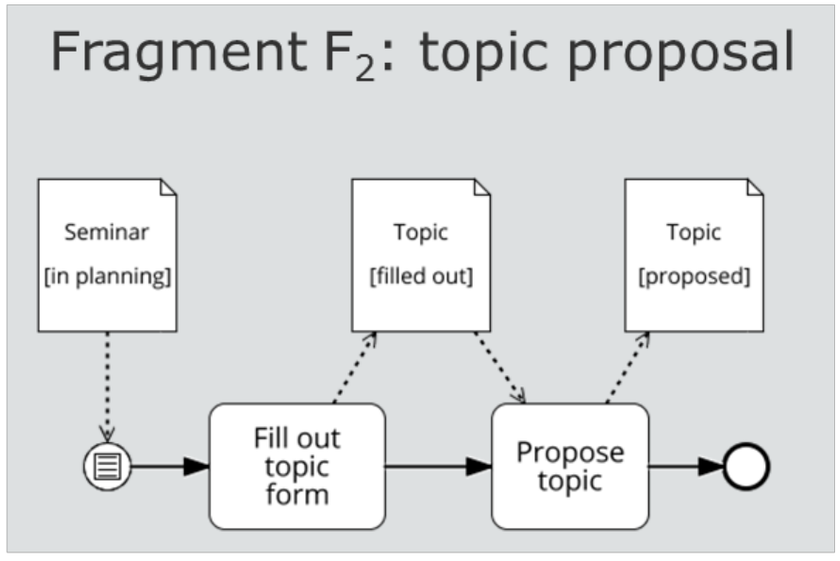


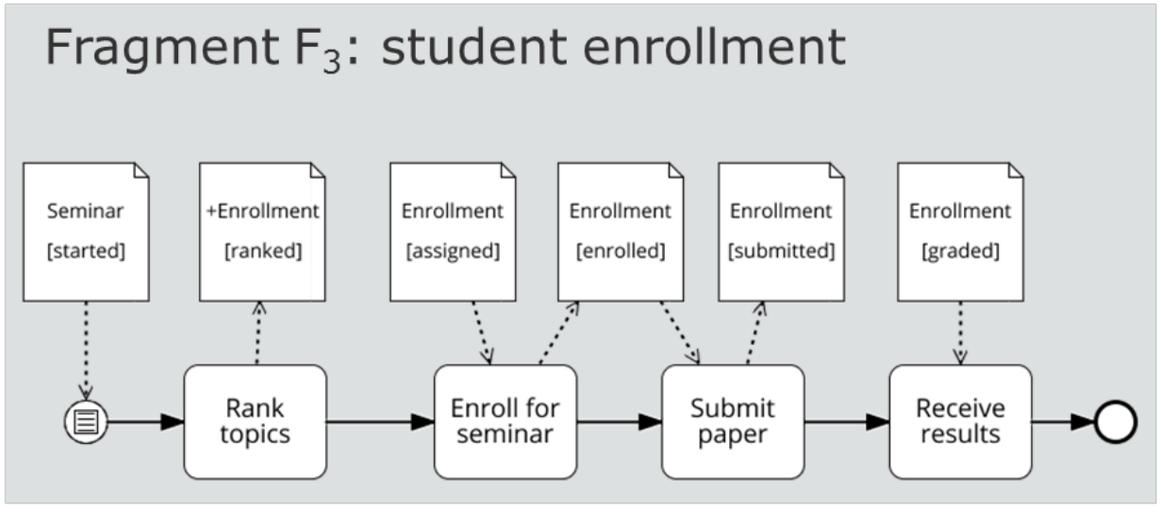
Chimera


initial
specify
in planing
evaluate
topics selected
start
started
cancel
not enough students
assign
topics assigned
grade
finished


Fragment F_1: seminar setup
+Seminar [inital]
Seminar [in planing]
Seminar [in planing]
topics
Topic [proposed]
Topic [selected]
topics
Topic [rejected]
-topics
Seminar [topics selected]
Select title & organizer
Specify seminar requirements
Ask staff for topic proposals
topic proposal deadline
Discuss and select seminar topics


<<case class>> Seminar
title :String
organizer :StaffMember [1..2]
topics :Topic [5..*]
students :Enrollment [3..15]
level :{BA, MA}
ECTS points :{3, 6}
1
3..15
students
seminar
Enrollment
name :String
email :String
matrikel :String
topic :Topic
grade :Grade
seminar :Seminar
1
organizer
1
topics
1..4
topic
1..2
1
assignee
StaffMember
name :String
email :String
position :String
5..*
Topic
title :String
supervisor :StaffMember [1..2]
assignee :Enrollment [1..4]
description :String
1..2
1
supervisor


Fragment F_2: topic proposal
Seminar [in planning]
Topic [filled out]
Topic [proposed]
Fill out topic form
Propose topic


Fragment F_3: student enrollment
Seminar [started]
+Enrollment [ranked]
Enrollment [assigned]
Enrollment [enrolled]
Enrollment [submitted]
Enrollment [graded]
Rank topics
Enroll for seminar
Submit paper
Receive results

Cooking with Business Entities

ARTIFACT-/OBJECT-CENTRIC PROCESSES

- ~ Explicit control-flow
- ~ Local, case data
- ✓ Global, persistent data
- ✓ Queries/updates on the persistent data
- ~ External inputs
- ~ Internal generation of fresh IDs

Back to the roots…

[PN15, Triebel and Sürmeli]
Algebraic PNs

[PN16, Lasota]
Survey on PNs with data

[ToPNoC17,_]
DB-Nets
(CPNs + DBs)

[CAiSE10, Sidorova et al.]
Conceptual nets

[BPM2013, De Leoni and van der Aalst]
DPNs

[FAOC16, _]
Verification of PNs with names

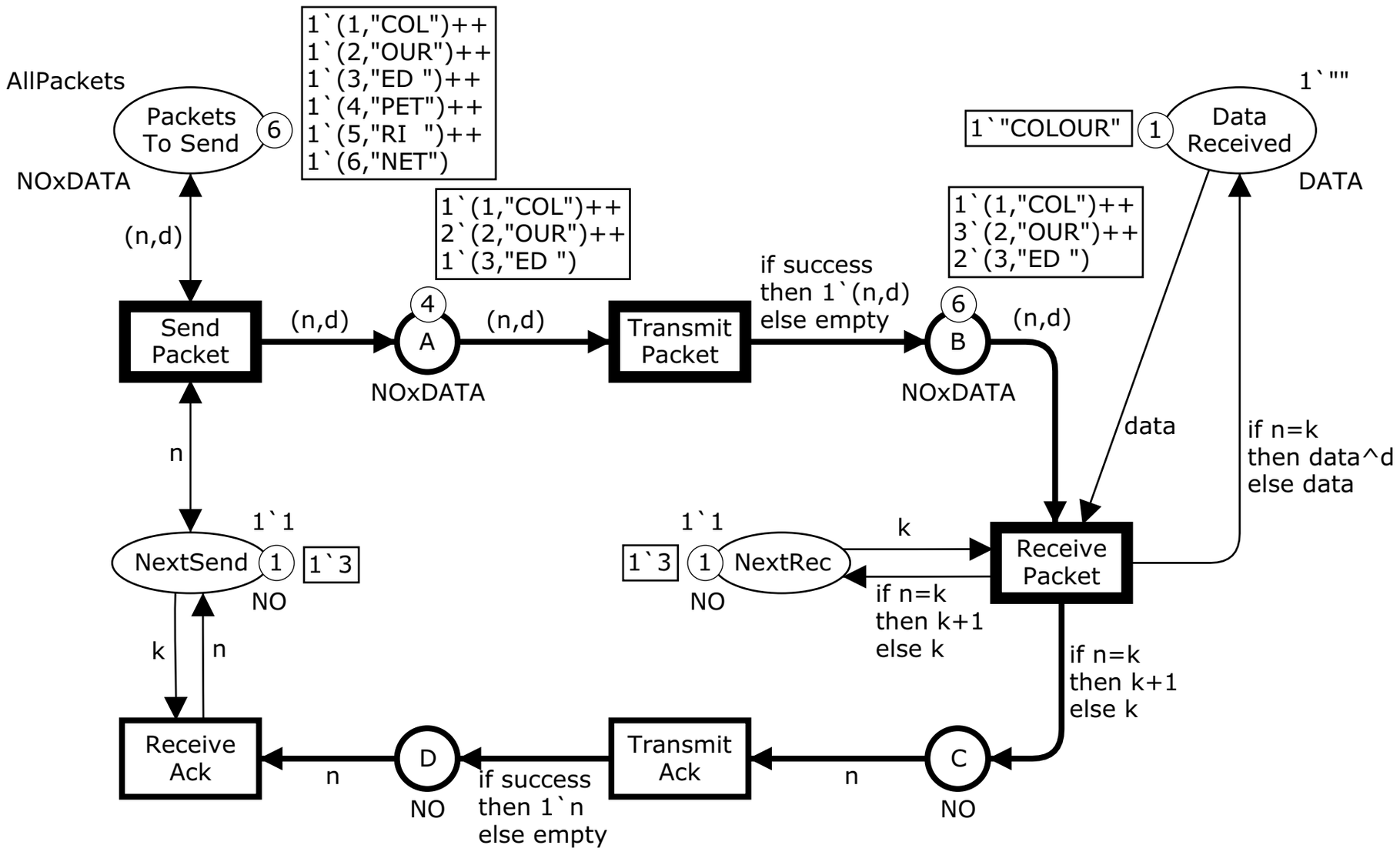
[ICATPN07, Lazic et al.]
Data Nets

...

[TCS11, Rosa-Velardo and de Frutos-Escrig]
v-PNs
(nets managing names)

[AAAI17, _]
RAW-SYS
(Workflow nets + DBs)

Colored Petri Nets



No **conceptual** representation of persistent storage

Recipe?

COLORED PETRI NETS

- ✓ Explicit control-flow
- ✓ Local, case data
- ✗ Global, persistent data
- ✗ Queries/updates on the persistent data
- ✓ External inputs
- ✓ Internal generation of fresh IDs

implicit, or using fresh variables

Verifiability as a requirement

[ICDT05, Vardi]
Model checking for database theoreticians

[PODS98, Abiteboul et al.]
Relational Transducers

[ICDT09, Vianu]
Verification of artifact-centric processes

[ECAI12, _]
Knowledge and action bases

[PODS13, Bojanczyk et al.]
Verification via amalgamation

[I&C17, _]
FO μ-Calculus over Generic Transition Systems

[AIJ16, De Giacomo et al.]
Bounded SitCalc Action Theories

data-centric

[PODS13, _]
Verification of data-centric processes

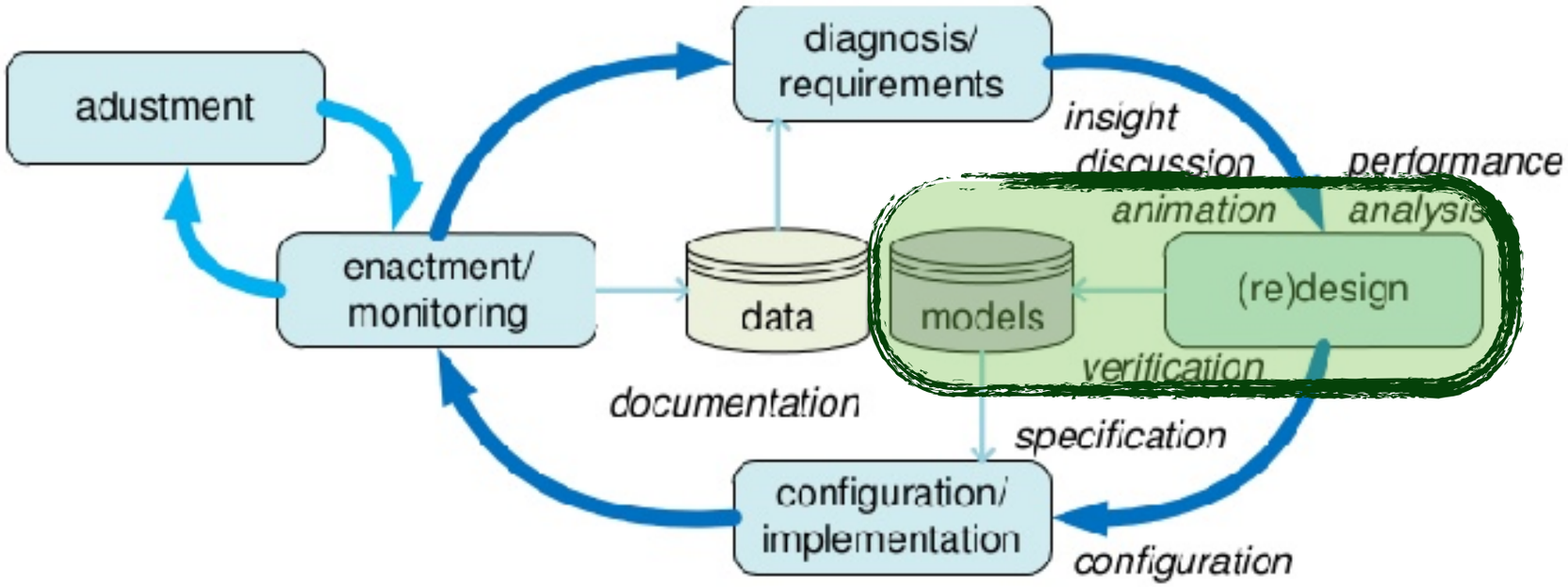
[PODS13, _]
Data-Centric Dynamic Systems

[PODS16, _]
Verification via under approximation

[STTT16, _]
Case-centric DCDS

1990 ... 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017

Formal Verification



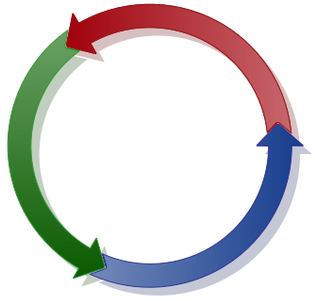
picture by Wil van der Aalst

Automated analysis
of a **formal model** of the system
against a property of interest,
considering **all** possible system behaviors

Formal Verification

The Conventional, Propositional Case

Process control-flow

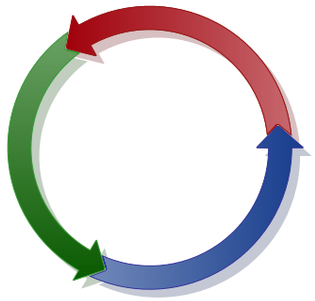


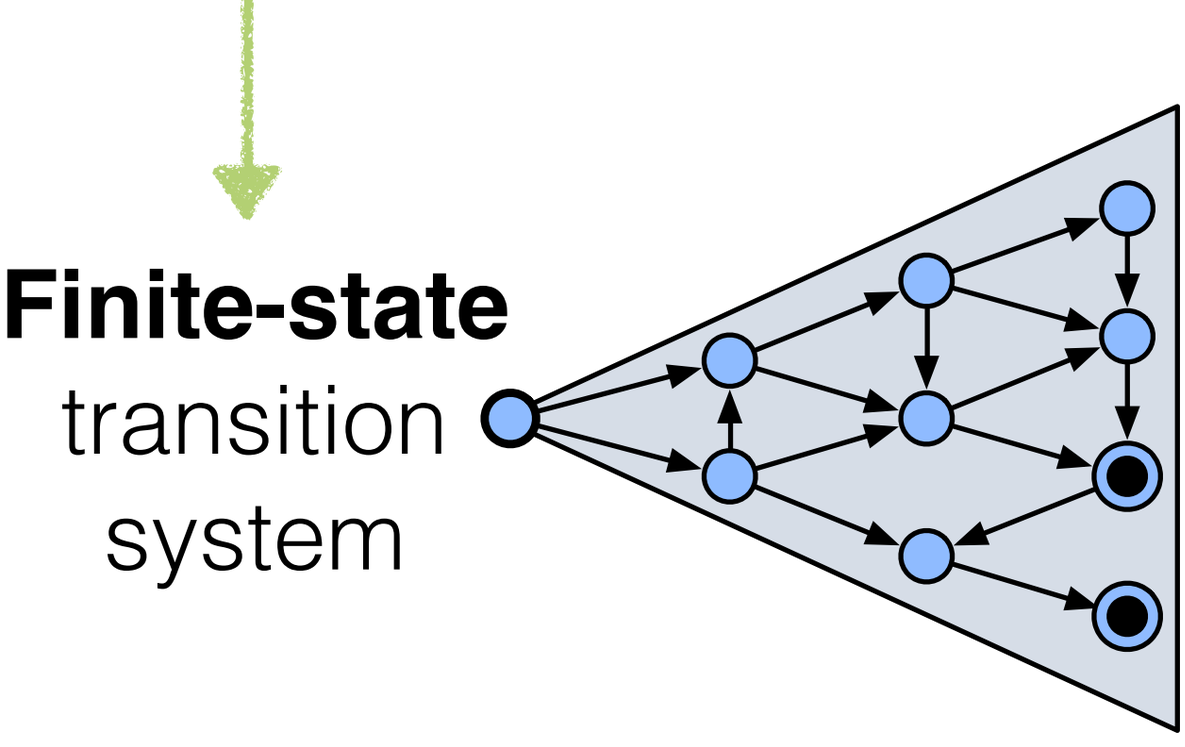
(Un)desired property

Formal Verification

The Conventional, Propositional Case

Process control-flow





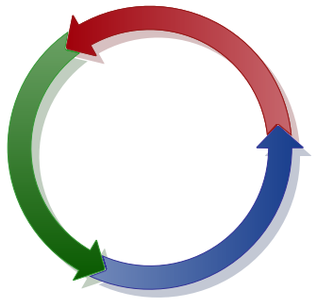
Φ **Propositional**
temporal formula

(Un)desired property

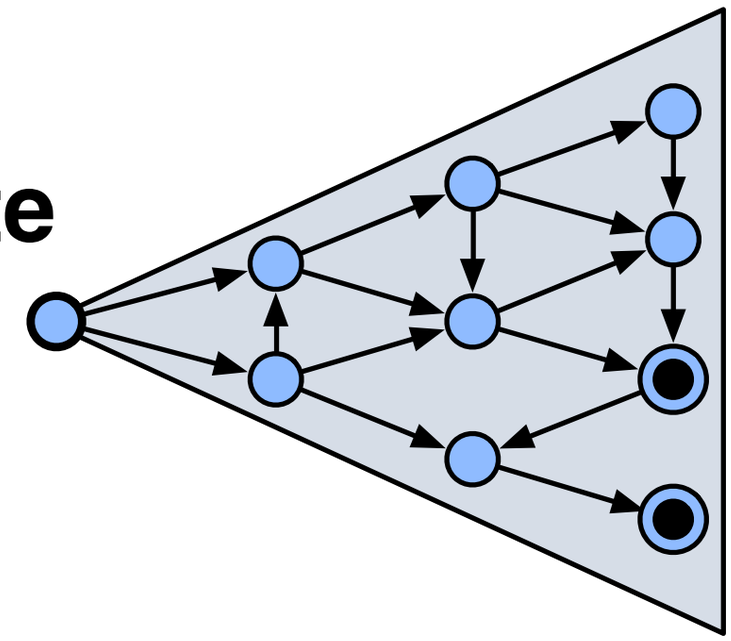
Formal Verification

The Conventional, Propositional Case

Process control-flow



Finite-state
transition
system



$\models \Phi$

Verification
via model checking
2007 Turing award:
Clarke, Emerson, Sifakis

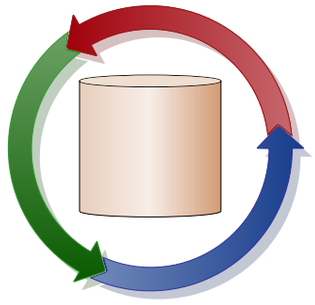
Propositional
temporal formula

(Un)desired property

Formal Verification

The Data-Aware Case

Data-aware process

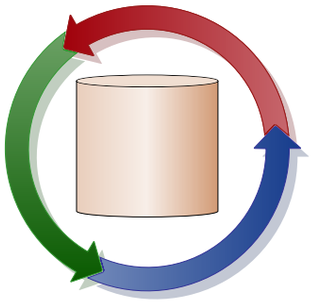


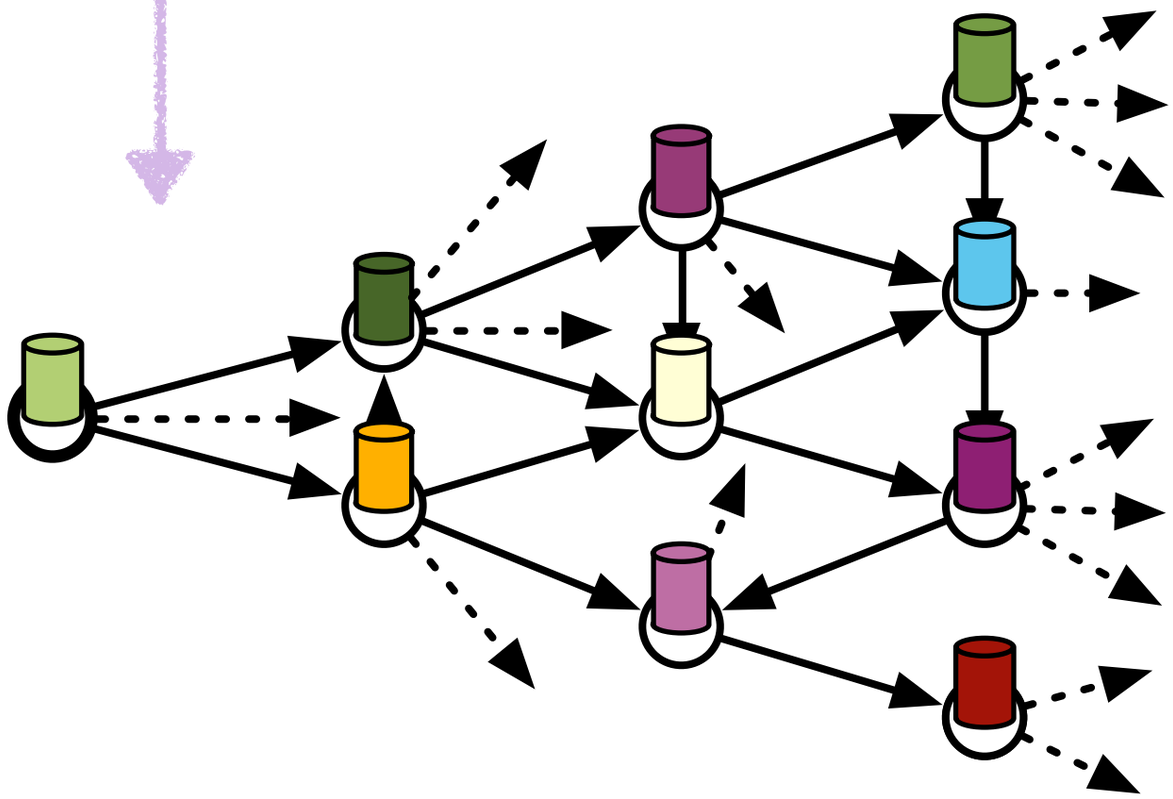
(Un)desired property

Formal Verification

The Data-Aware Case

Data-aware process





Infinite-state, relational
transition system [Vardi 2005]

Φ

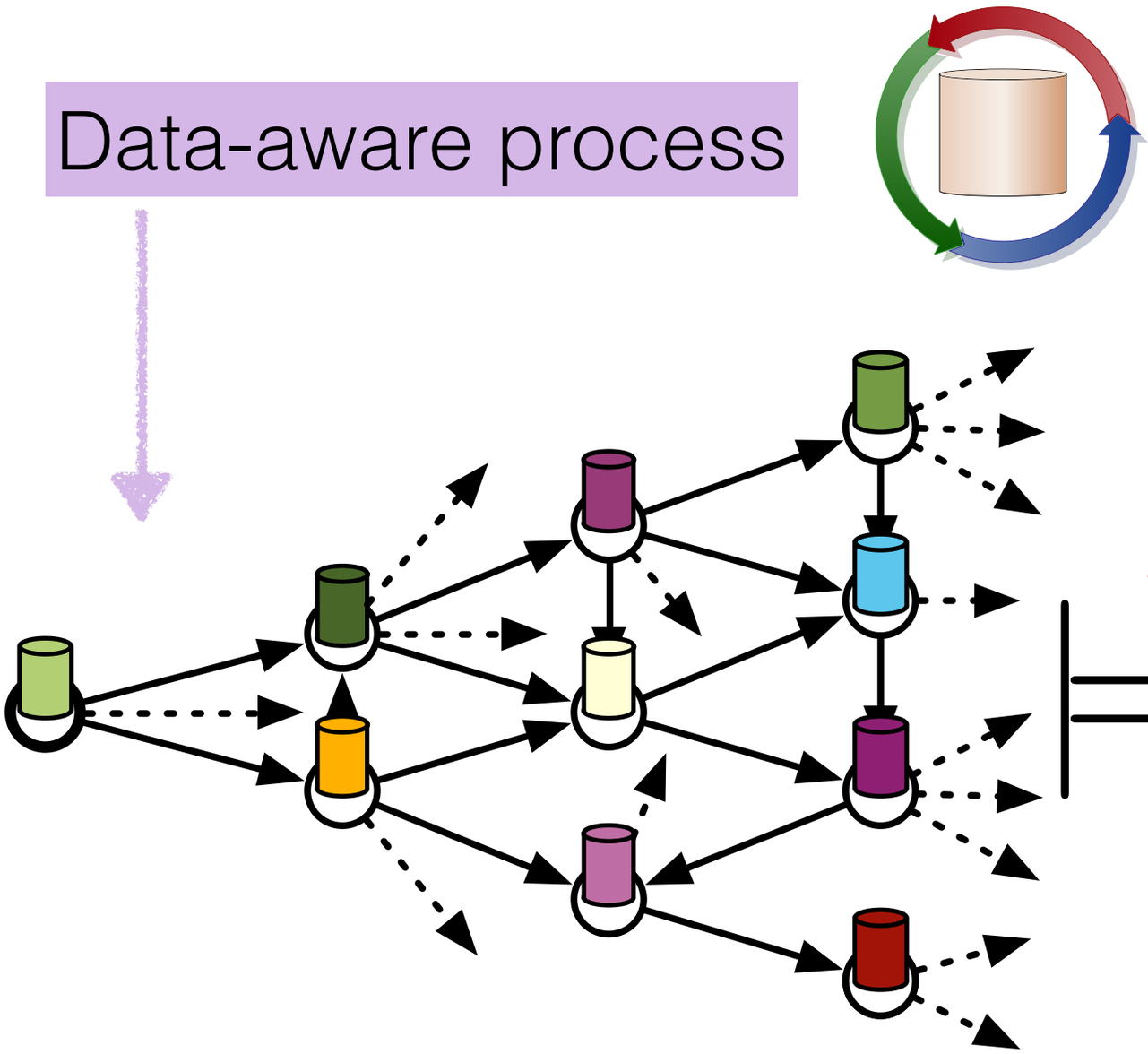
First-order
temporal formula

(Un)desired property

Formal Verification

The Data-Aware Case

Data-aware process



Infinite-state, relational
transition system [Vardi 2005]

$\models \Phi$

First-order
temporal formula

(Un)desired property

Why FO Temporal Logics

- To inspect **data**: **FO queries**
- To capture system **dynamics**: **temporal modalities**
- To track the **evolution of objects**: FO **quantification across** states
- Example: It is **always** the case that **every order is eventually** either **cancelled**, or **paid** and **then delivered**
- N.B.: the interplay between FO quantification and temporal modalities is quite subtle!

Problem Dimensions

Dimension 1
Static Information Model

How are data structured?

- Propositional symbols —> Finite state system

- Fixed number of values from an unbounded domain

- Full-fledged database:
  - relational database
  - tree-structured data, XML
  - graph-structured data

Dimension 1
Static Information Model

Are constraints present? How are they interpreted?

- Complete data
- Data under incomplete information
  - ontology (with intensional part typically fixed)
  - full-fledged ontology-based data access system
- Hard vs soft-constraints (inconsistency-tolerance)

Dimension 2
Dynamic Component

- Implicit representation of time vs. implicit progression mechanism vs. explicit process

- When an explicit process is present:
  - how is the process dynamics represented?
  - procedural vs. declarative approaches (e.g., finite state machines vs. rule-based)

- Deterministic vs. non-deterministic behaviour

- Linear time vs. branching time model

- Finite vs. infinite traces

Dimension 3
Data-Process Interaction

How are data manipulated by the process?

- Data is only accessed, but not modified
- Data are updated, but no new values are inserted
- Full-fledged combination of the temporal and structural dimensions
- Hybrid approaches (e.g., read-only database + read-write registers)

Dimension 4
Interaction with the Environment

Is the system interacting with the external world?

- Closed systems vs. bounded input vs. unbounded input
- Synchronous vs. asynchronous communication
- Message passing, possibly with queues
- One-way or two-way service calls

Dimension 4
Interaction with the Environment

Which parts of the environment are fixed? Which change?

- Stateless vs stateful environment
- Fixed database vs. varying database vs. varying portion of data
- Multiple devices/agents interacting with each other
- Fixed vs changing topologies

Dimension 5
Formal Analysis

How are (un)desired properties formulated?

- Analysis of fundamental properties: reachability, absence of deadlock, boundedness, (weak) soundness

- Analysis of arbitrary formulae in some temporal logic

- Analysis of properties with queries across the temporal dimension (in the style of temporal DBs)

Dimension 5
Formal Analysis

Which forms of analysis?

- Verification
- Dominance, simulation, equivalence
- Synthesis from a given specification
- Composition of available components

1) Go to the essential
2) Find boundaries of decidability in a general setting
3) Understand the connection with concrete languages
4) Implement