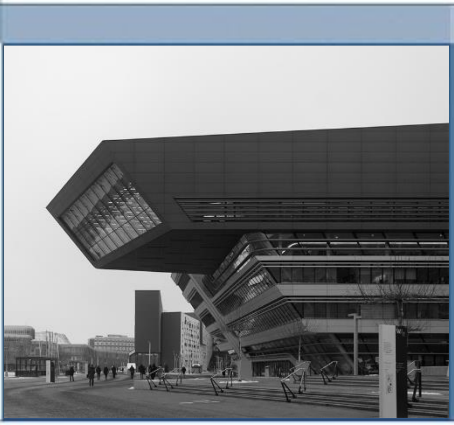
CAiSE Forum 2017

Essen – 15th June 2017

POLITECNICO DI MILANO

WIRTSCHAFTS
UNIVERSITÄT
WIEN VIENNA
UNIVERSITY OF
ECONOMICS
AND BUSINESS



Giovanni Meroni, Claudio Di Ciccio, Jan Mendling

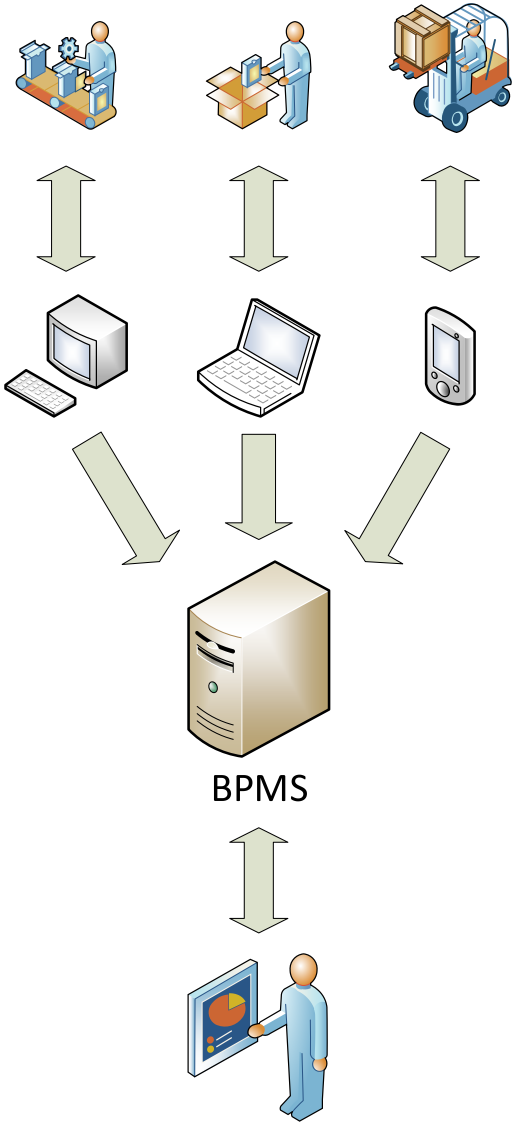
# ARTIFACT-DRIVEN PROCESS MONITORING: DYNAMICALLY BINDING REAL-WORLD OBJECTS TO RUNNING PROCESSES

# Context

- Most business processes involve multiple stakeholders
  - I.e.: freight transportation, supply chain, etc…
- Stakeholders control only a portion of the process
  - Can only observe the execution of other portions
- Real-world objects are manipulated during the execution
  - They belong to one stakeholder, yet are altered by everybody
  - For execution, activities require objects in a certain condition
  - Activities may alter the objects
- We call these objects **artifacts**

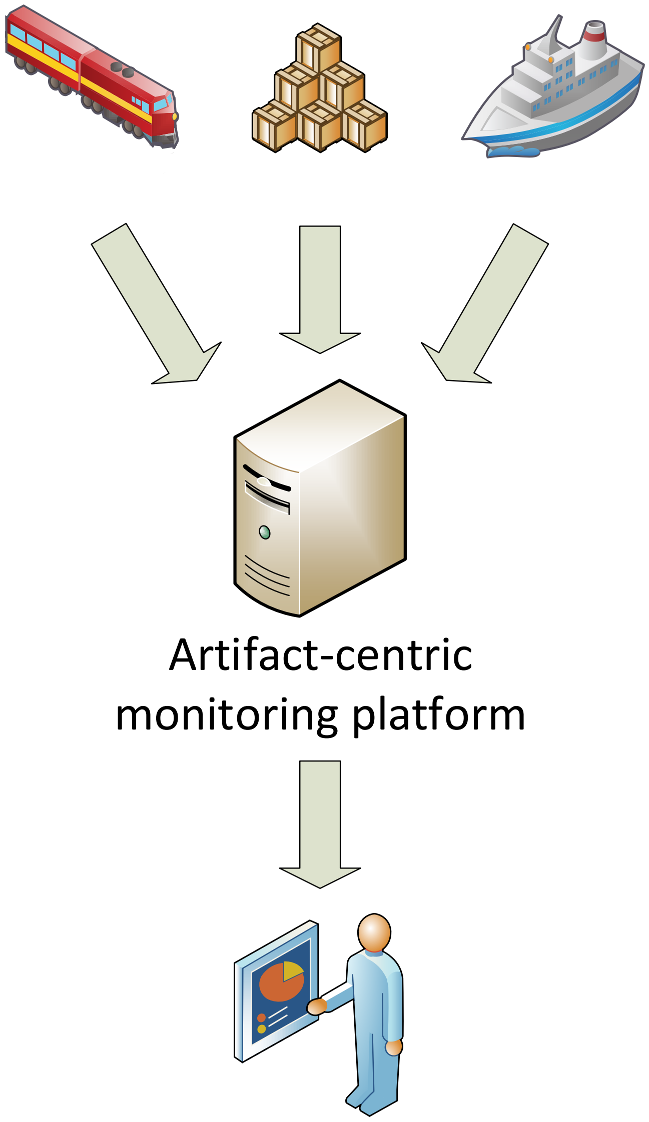
# Limitations of traditional approaches

- Stakeholders cannot be sure that the process will be executed as planned
  - They cannot enforce the execution of the whole process
  - Unforeseen exceptions may arise
  - Violations in the execution flow may occur
- Traditional BPMSs require human intervention when monitoring multi-party processes
  - The BPMS expects explicit notification when activities start or end
  - When not automated, notifications must be sent manually
  - When a violation in the execution occurs, the BPMS stops until the issue is solved



# Artifact-driven process monitoring
## Idea

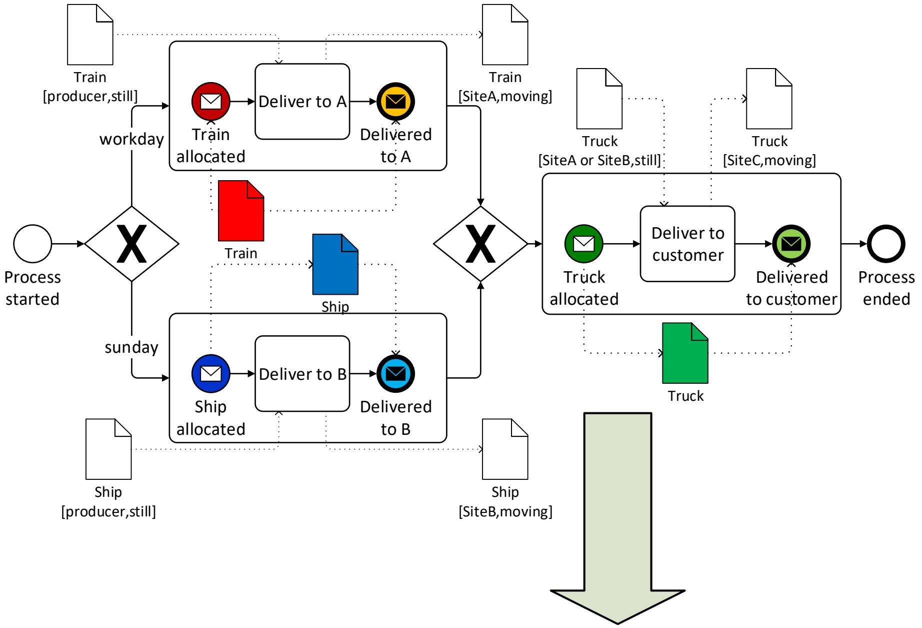
- Use artifacts to detect when activities are executed
  - The artifact “knows” when is altered
  - The artifact “knows” when activities are executed
- Artifact-centric languages better at monitoring the process
  - Dependencies among activities are descriptive, not prescriptive
  - Even after violations, monitoring continues

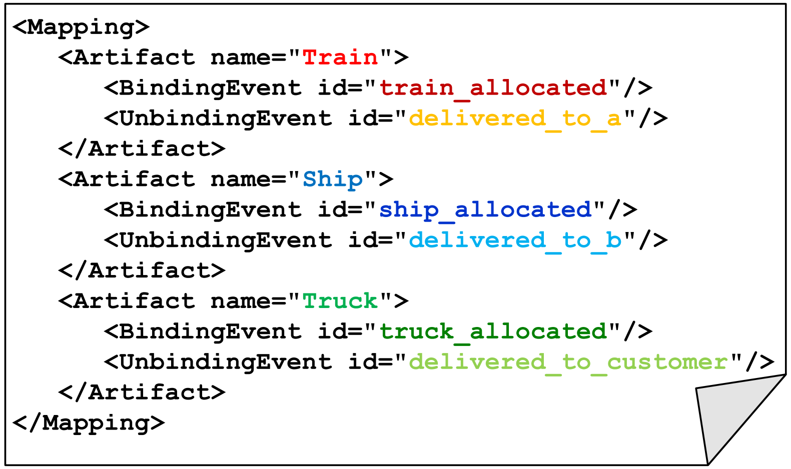


Artifact-centric monitoring platform

# Artifact-driven process monitoring
## Solution

- Artifacts are bound to process instances at runtime
  - Their identity is known after the process started
  - Artifacts may participate to a portion of the process
  - Artifacts may participate to multiple processes simultaneously
- Mechanisms to dynamically bind artifacts are provided





```
<Mapping>
   <Artifact name="Train">
      <BindingEvent id="train_allocated"/>
      <UnbindingEvent id="delivered_to_a"/>
   </Artifact>
   <Artifact name="Ship">
      <BindingEvent id="ship_allocated"/>
      <UnbindingEvent id="delivered_to_b"/>
   </Artifact>
   <Artifact name="Truck">
      <BindingEvent id="truck_allocated"/>
      <UnbindingEvent id="delivered_to_customer"/>
   </Artifact>
</Mapping>
```