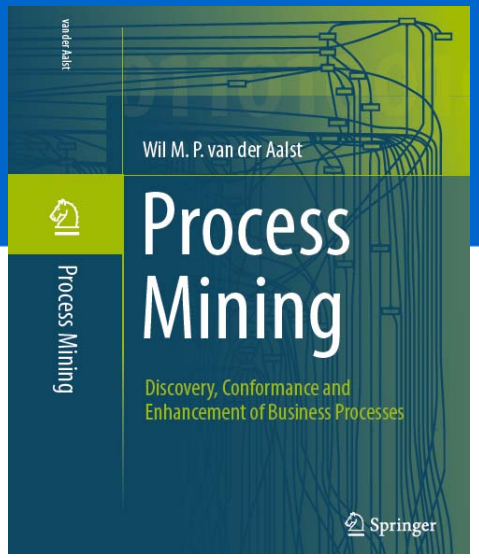
Chapter 9
Operational Support

prof.dr.ir. Wil van der Aalst
www.processmining.org



TU/e Technische Universiteit Eindhoven University of Technology

Where innovation starts

Overview

Chapter 1
Introduction

Part I: Preliminaries

Chapter 2
Process Modeling and Analysis

Chapter 3
Data Mining

Part II: From Event Logs to Process Models

Chapter 4
Getting the Data

Chapter 5
Process Discovery: An Introduction

Chapter 6
Advanced Process Discovery Techniques

Part III: Beyond Process Discovery

Chapter 7
Conformance Checking

Chapter 8
Mining Additional Perspectives

Chapter 9
Operational Support

Part IV: Putting Process Mining to Work

Chapter 10
Tool Support

Chapter 11
Analyzing “Lasagna Processes”

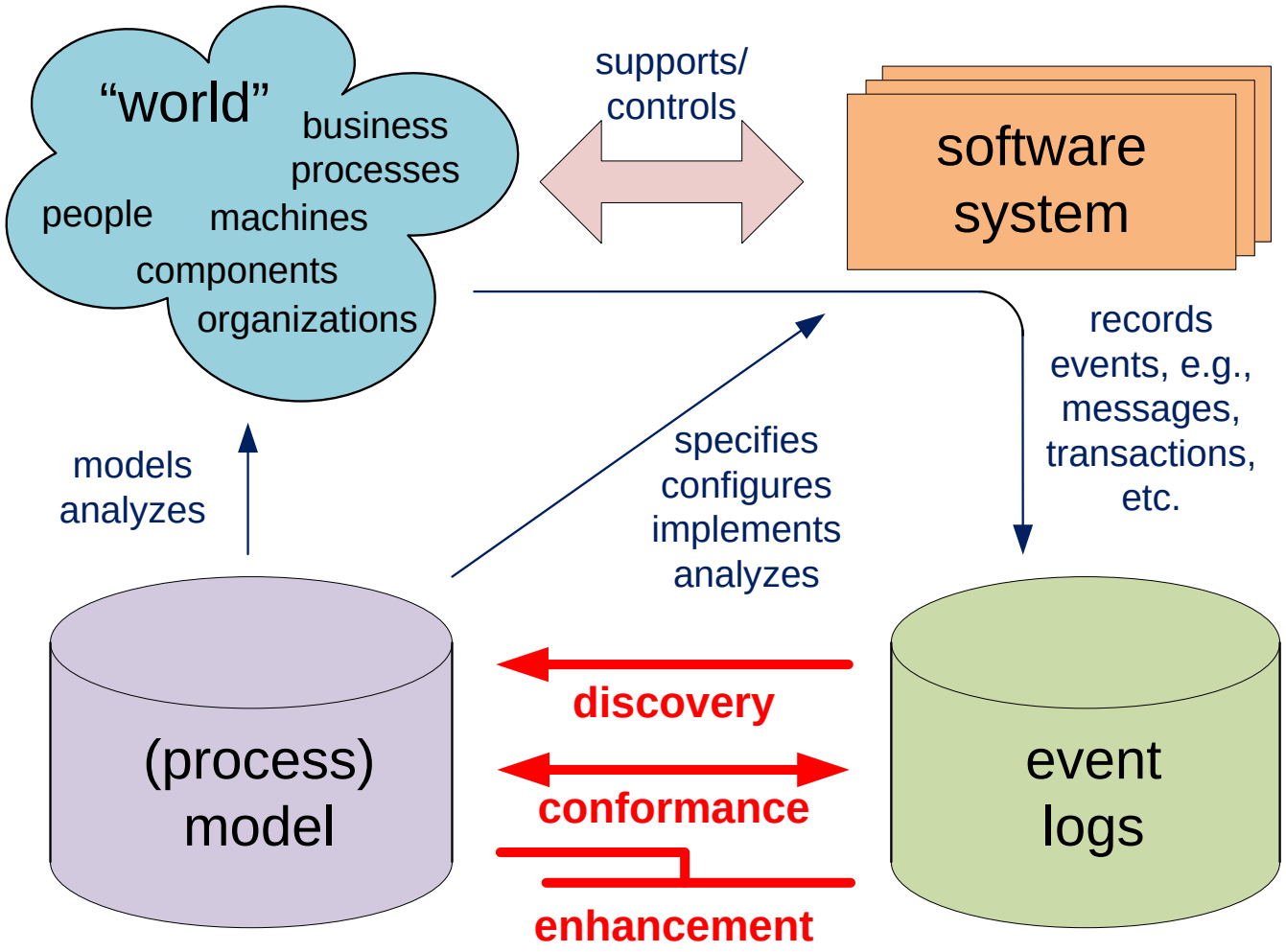
Chapter 12
Analyzing “Spaghetti Processes”

Part V: Reflection

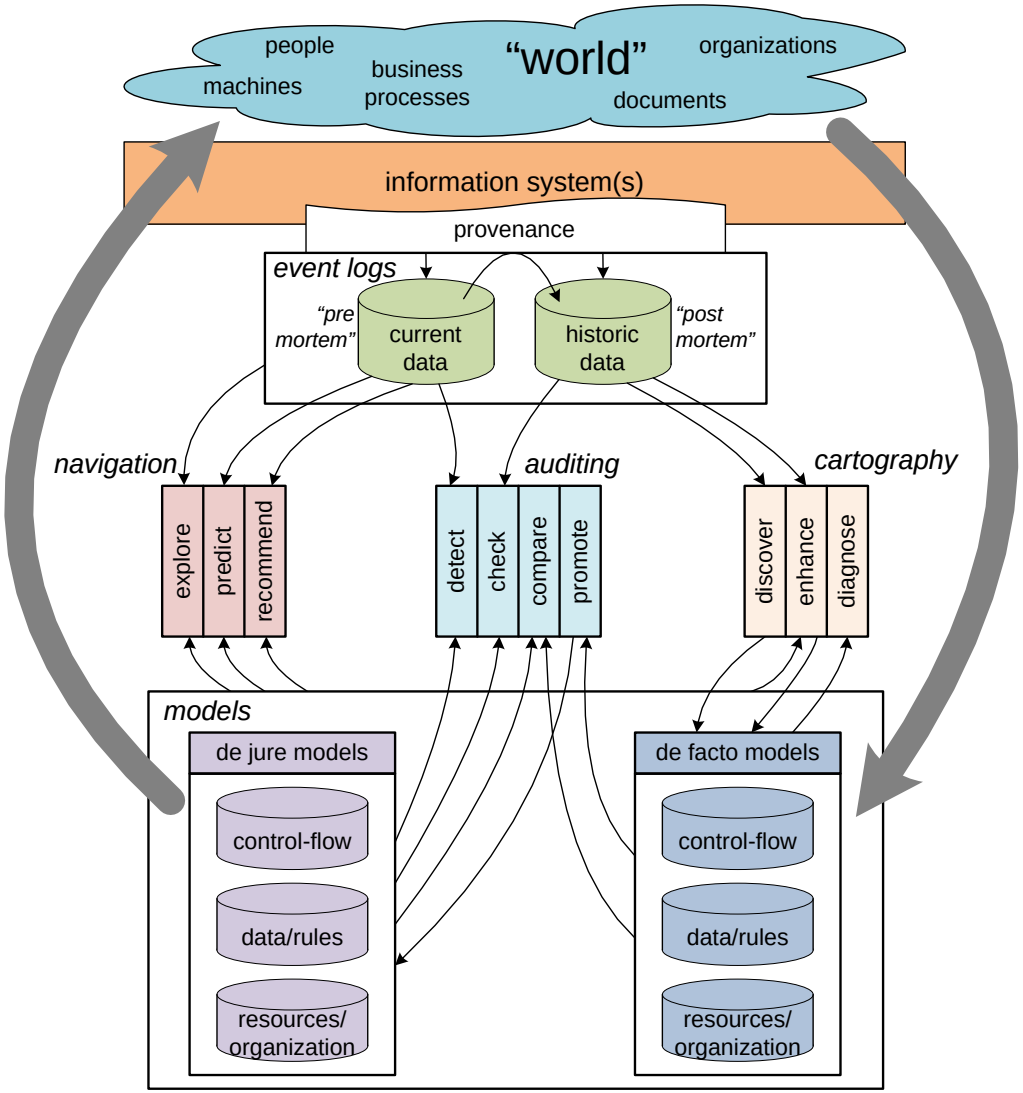
Chapter 13
Cartography and Navigation

Chapter 14
Epilogue

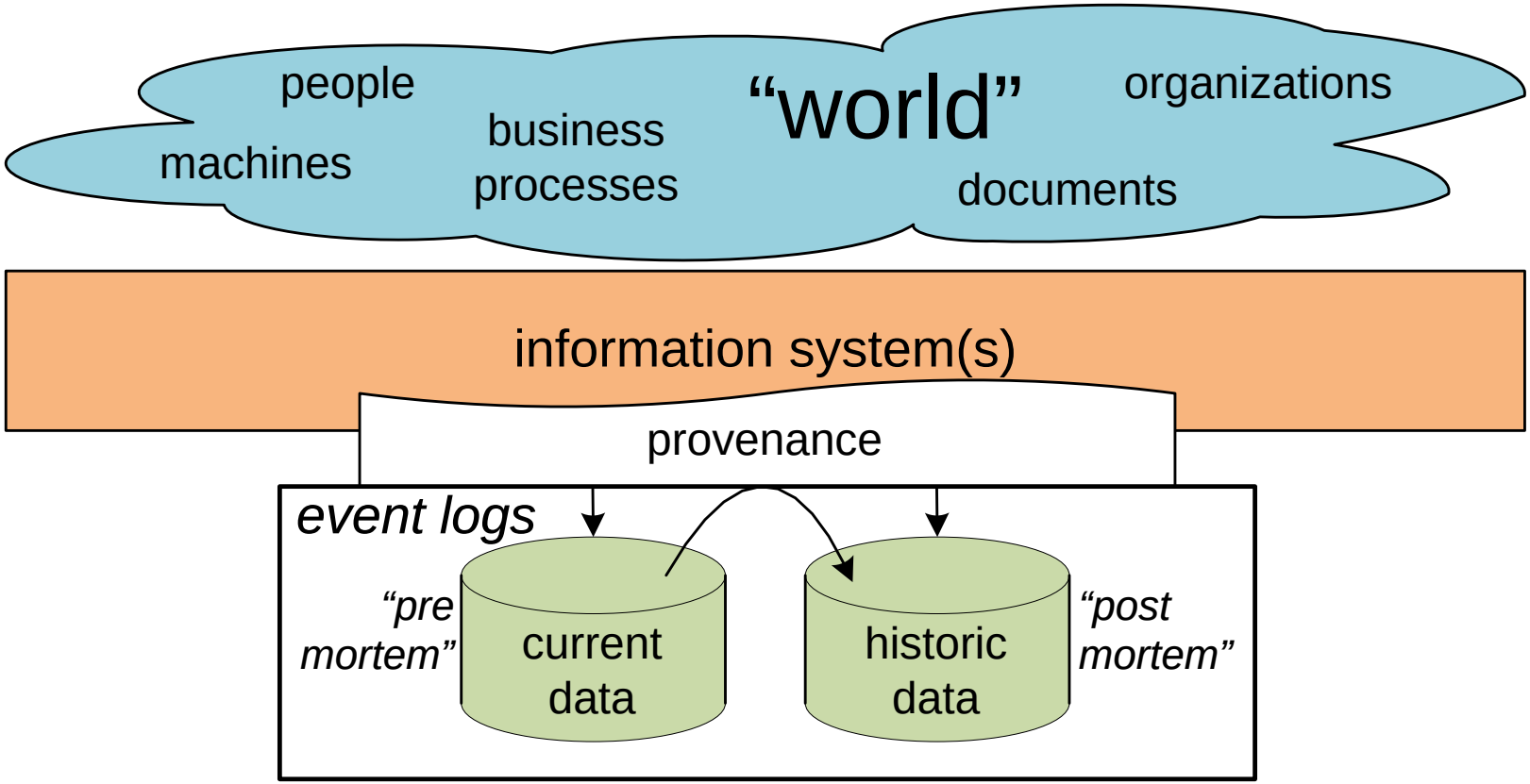
Process mining spectrum



Refined process mining framework



Business process provenance



Two types of event data: post and pre mortem

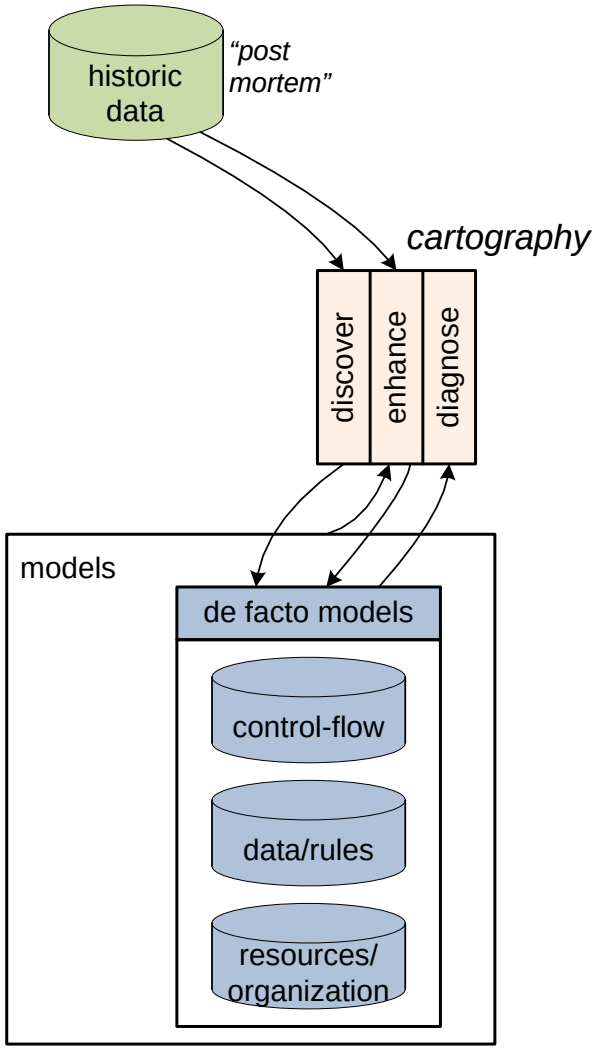
- **“Post mortem” event data refer to information about cases that have completed, i.e., these data can be used for process improvement and auditing, but not for influencing the cases they refer to.**
- **“Pre mortem” event data refer to cases that have not yet completed. If a case is still running, i.e., the case is still “alive” (pre mortem), then it may be possible that information in the event log about this case (i.e., current data) can be exploited to ensure the correct or efficient handling of this case.**

Two types of models: “de jure models” and “de facto models”

- A de jure model is normative, i.e., it specifies how things should be done or handled. For example, a process model used to configure a BPM system is normative and forces people to work in a particular way.
- A de facto model is descriptive and its goal is not to steer or control reality. Instead, de facto models aim to capture reality.

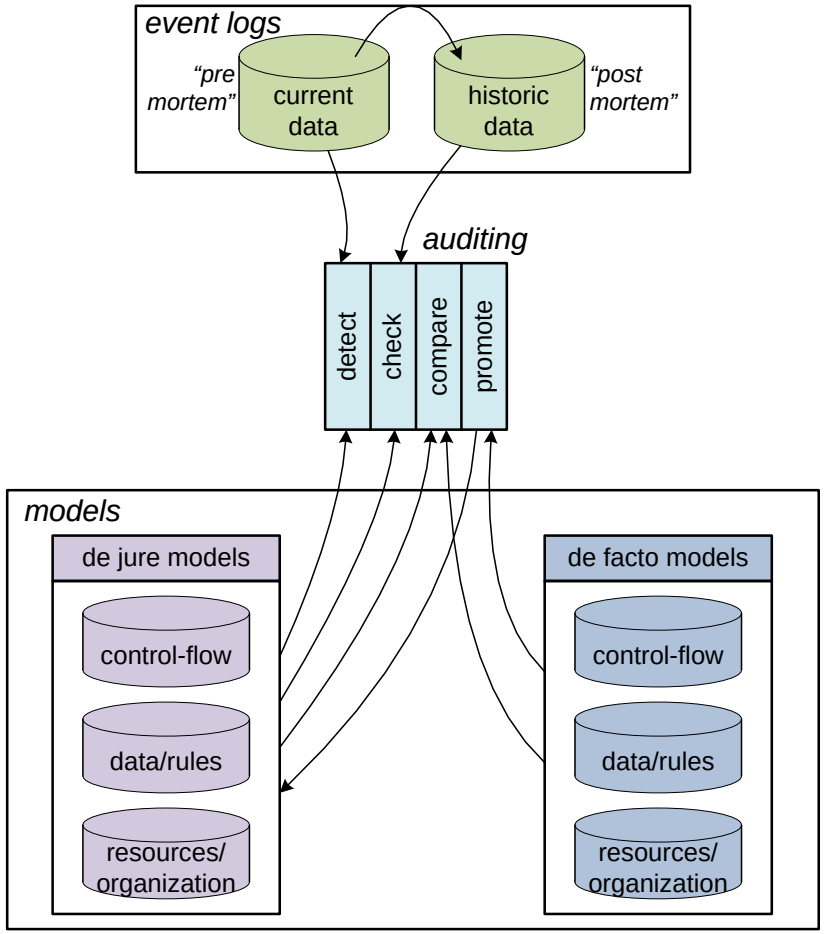
Cartography

- **Discover. This activity is concerned with the extraction of (process) models.**
- **Enhance. When existing process models (either discovered or hand-made) can be related to events logs, it is possible to enhance these models.**
- **Diagnose. This activity does not directly use event logs and focuses on classical model-based analysis.**



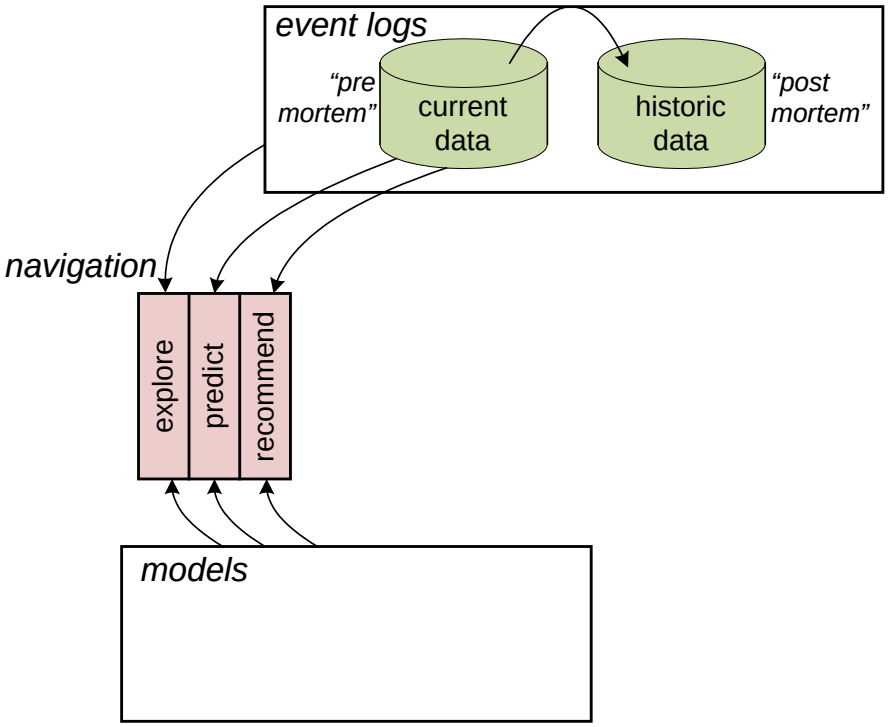
Auditing

- **Detect. Compares de jure models with current "pre mortem" data. The moment a predefined rule is violated, an alert is generated (online).**
- **Check. The goal of this activity is to pinpoint deviations and quantify the level of compliance (offline).**
- **Compare. De facto models can be compared with de jure models to see in what way reality deviates from what was planned or expected.**
- **Promote. Promote parts of the de facto model to a new de jure model.**

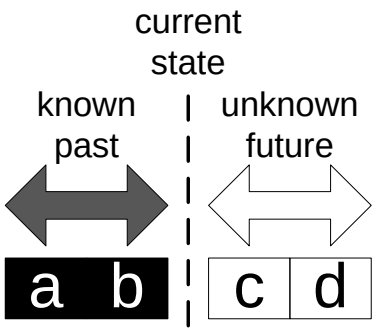


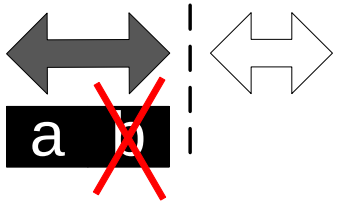
Navigation

- **Explore. The combination of event data and models can be used to explore business processes at run-time.**
- **Predict. By combining information about running cases with models, it is possible to make predictions about the future, e.g., the remaining flow time and the probability of success.**
- **Recommend. The information used for predicting the future can also be used to recommend suitable actions (e.g. to minimize costs or time).**

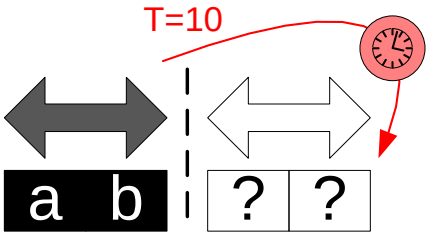


Operational support:
online process mining using “pre mortem” event data

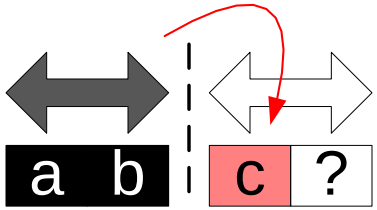




***detect**: b does not fit the model (not allowed, too late, etc.)*



***predict**: some prediction is made about the future (e.g. completion date or outcome)*

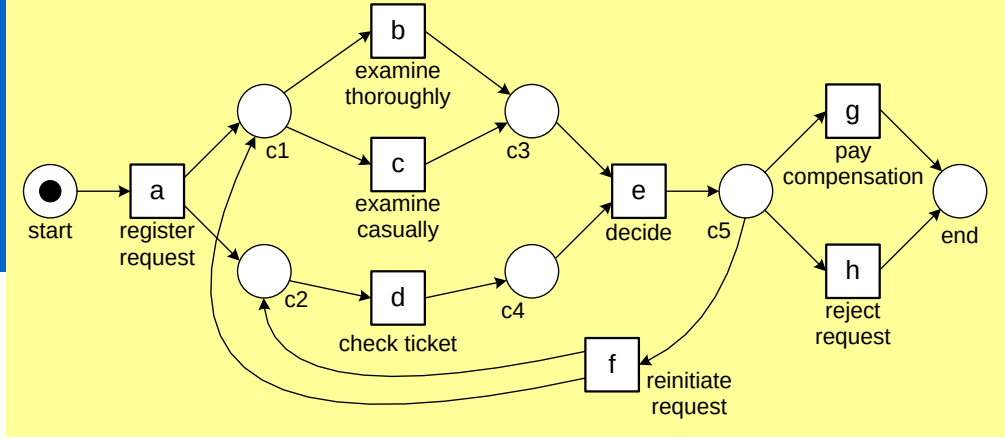


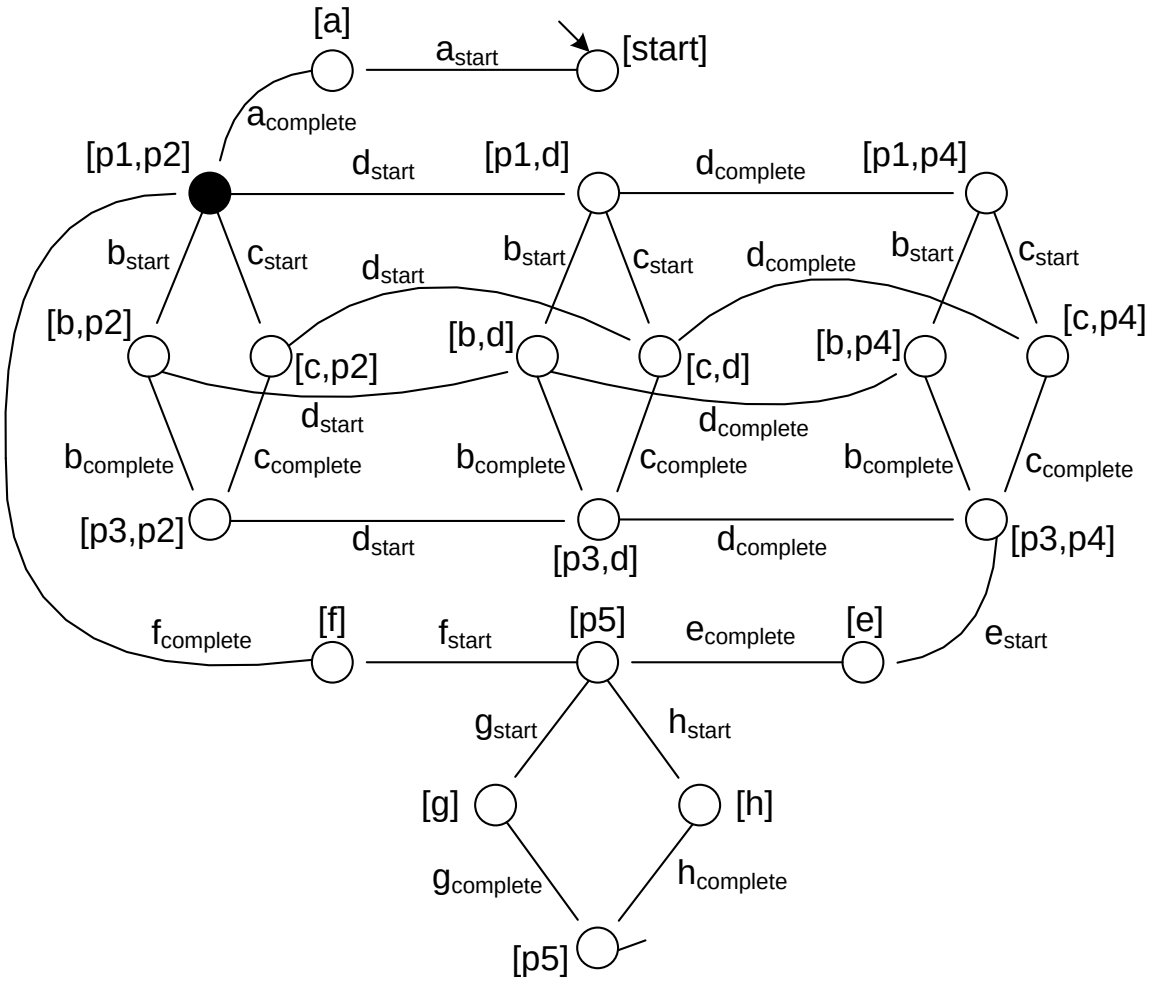
***recommend**: based on past experiences c is recommended (e.g., to minimize costs)*

Let us focus one time

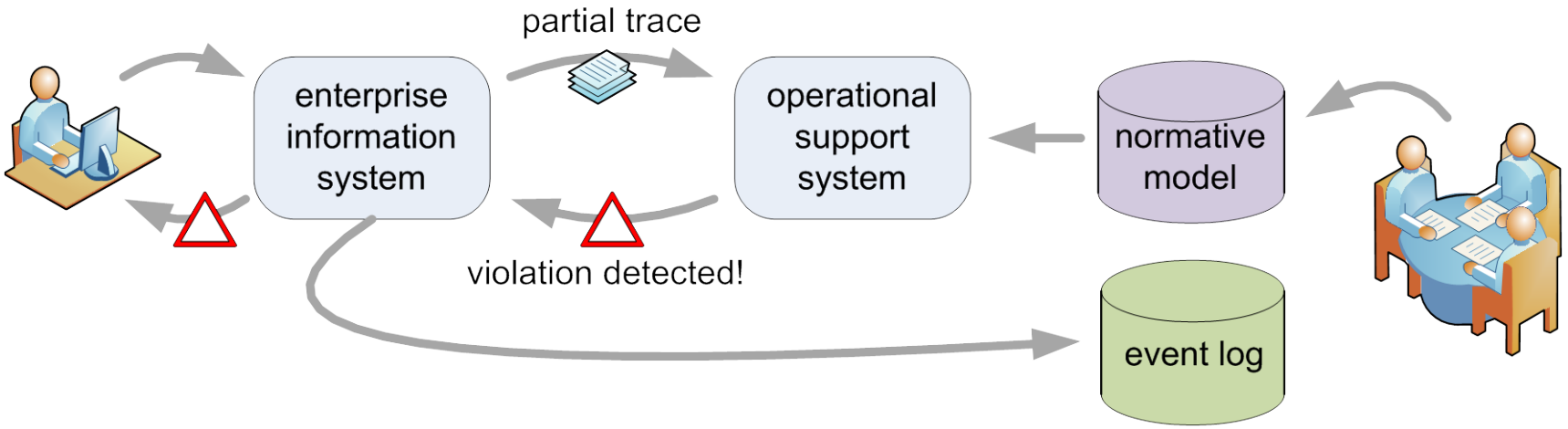
case id	trace
1	$\langle a^{12}_{start}, a^{19}_{complete}, b^{25}_{start}, d^{26}_{start}, b^{32}_{complete}, d^{33}_{complete}, e^{35}_{start}, e^{40}_{complete}, h^{50}_{start}, h^{54}_{complete}\rangle$
2	$\langle a^{17}_{start}, a^{23}_{complete}, d^{28}_{start}, c^{30}_{start}, d^{32}_{complete}, c^{38}_{complete}, e^{50}_{start}, e^{59}_{complete}, g^{70}_{start}, g^{73}_{complete}\rangle$
3	$\langle a^{25}_{start}, a^{30}_{complete}, c^{32}_{start}, c^{35}_{complete}, d^{35}_{start}, d^{40}_{complete}, e^{45}_{start}, e^{50}_{complete}, f^{50}_{start}, f^{55}_{complete}, b^{60}_{start}, d^{62}_{start}, b^{65}_{complete}, d^{67}_{complete}, e^{80}_{start}, e^{87}_{complete}, g^{90}_{start}, g^{98}_{complete}\rangle$
...	...

Transition system (with start/complete)





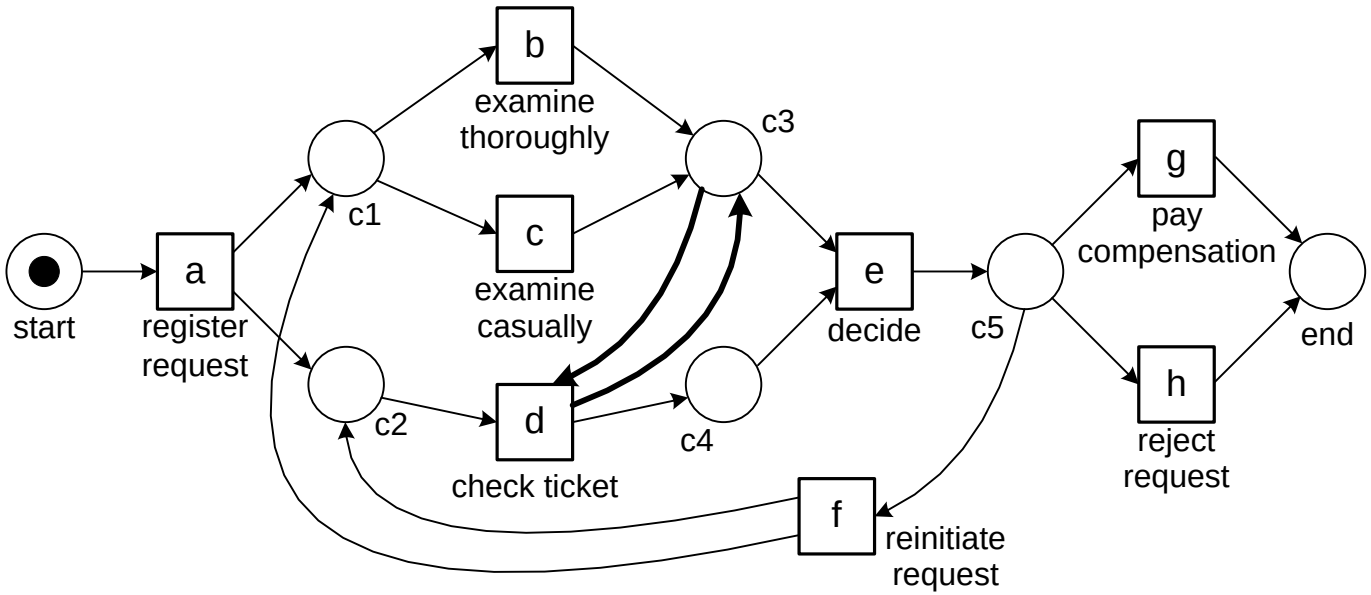
Operational support: Detect



Example

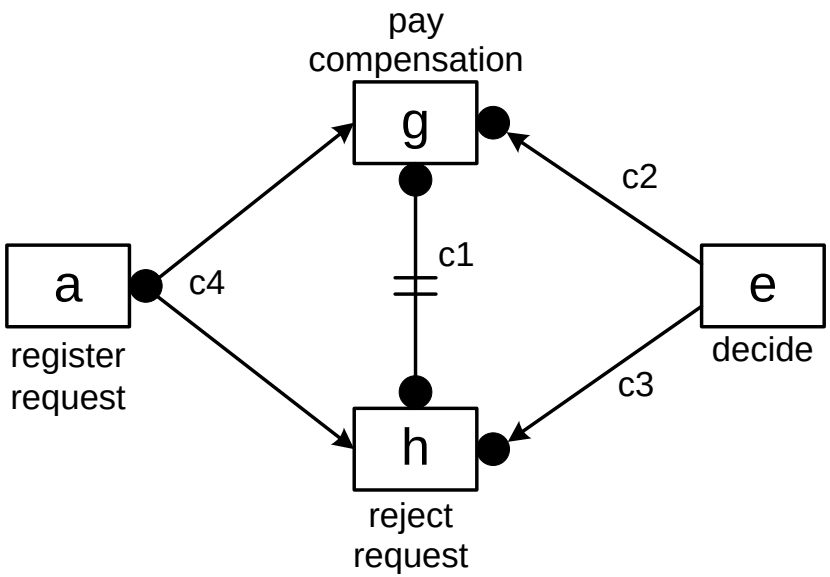
$$\langle a^{12}_{start}, a^{19}_{complete}, b^{25}_{start}, d^{26}_{start} \rangle$$

alert!!!!



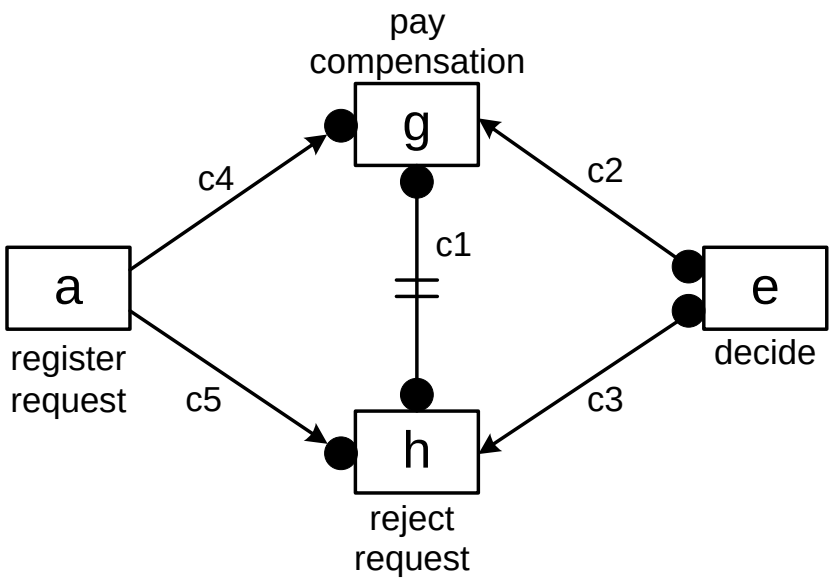
case id	trace
1	$\langle a^{12}_{start}, a^{19}_{complete}, b^{25}_{start}, d^{26}_{start}, b^{32}_{complete}, d^{33}_{complete}, e^{35}_{start}, e^{40}_{complete}, h^{50}_{start}, h^{54}_{complete} \rangle$
2	$\langle a^{17}_{start}, a^{23}_{complete}, d^{28}_{start}, d^{32}_{complete}, c^{38}_{complete}, e^{50}_{start}, e^{59}_{complete}, g^{70}_{start}, g^{73}_{complete} \rangle$
3	$\langle a^{25}_{start}, a^{30}_{complete}, c^{35}_{start}, c^{35}_{complete}, d^{35}_{start}, d^{40}_{complete}, e^{45}_{start}, e^{50}_{complete}, f^{50}_{start}, f^{55}_{complete}, b^{60}_{start}, d^{62}_{start}, b^{65}_{complete}, d^{67}_{complete}, e^{80}_{start}, e^{87}_{complete}, g^{90}_{start}, g^{98}_{complete} \rangle$
...	...

Declare specifications for detecting violations



- **Satisfied: the LTL formula evaluates to true for the current partial trace.**
- **Temporarily violated: the LTL formula evaluates to false, however, there is a longer trace that evaluates to true.**
- **Permanently violated: the LTL formula evaluates to false for current trace and all its extensions**

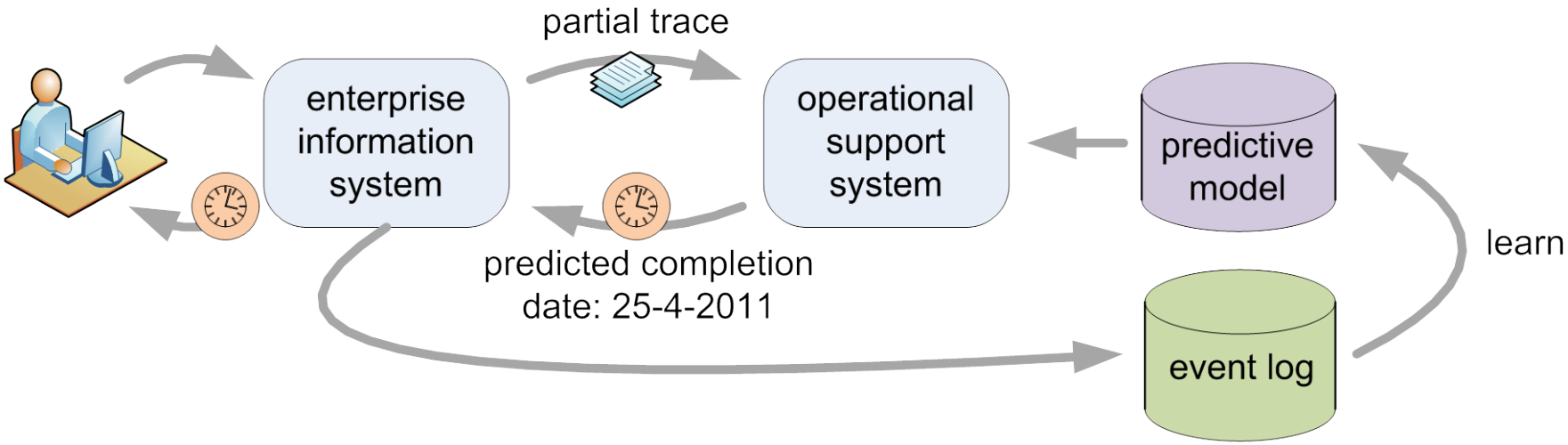
Conflicting constraints



- A Declare specification is satisfied for a case if all of its constraints are satisfied.
- A Declare specification is temporarily violated by a case if for the current partial trace at least one of the constraints is violated, however, there is a possible future in which all constraints are satisfied.
- A Declare specification is permanently violated by a case if no such future exists.

Note that c1, c2, and c3 imply that e cannot be executed without permanently violating the specification.

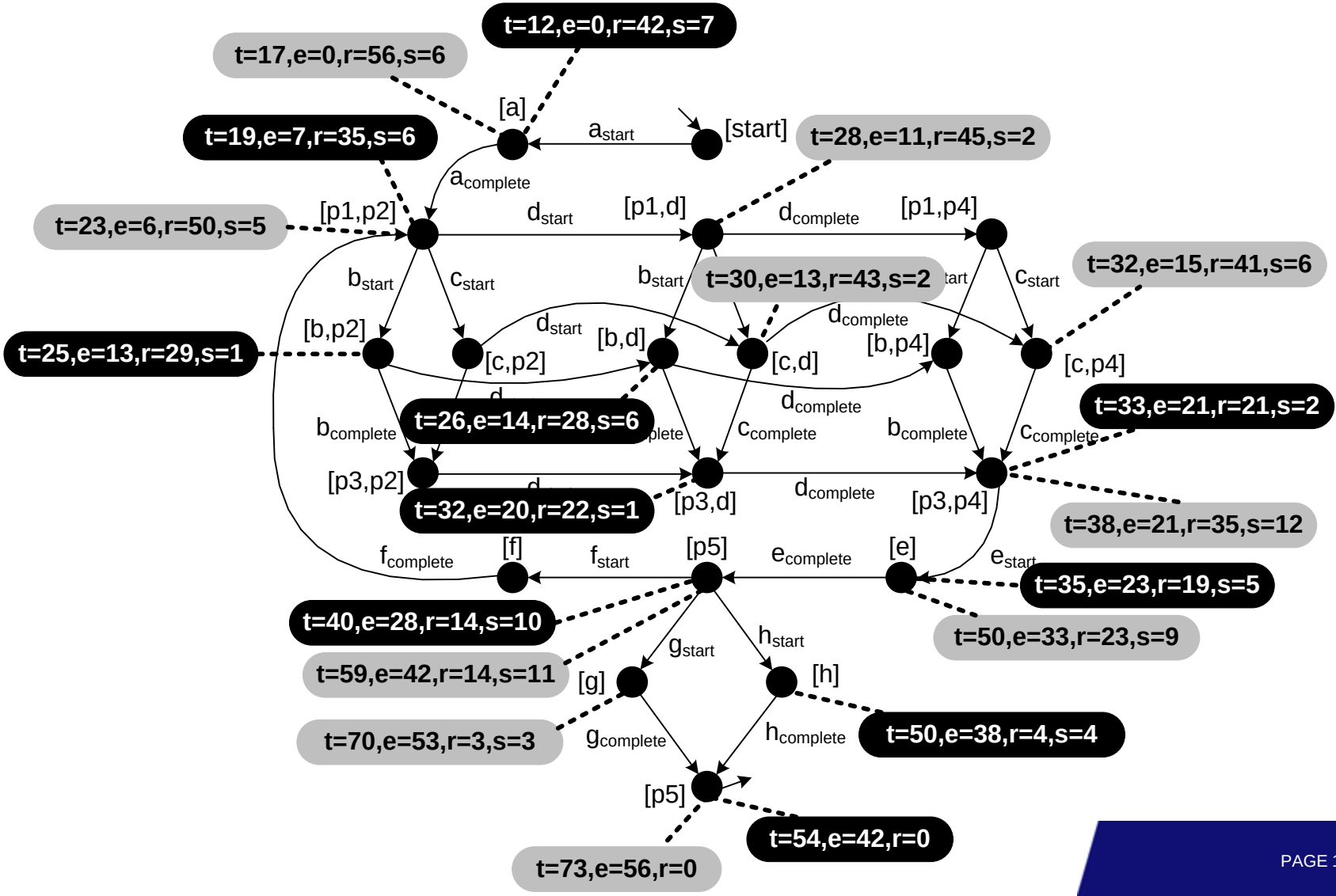
Operational support: Predict



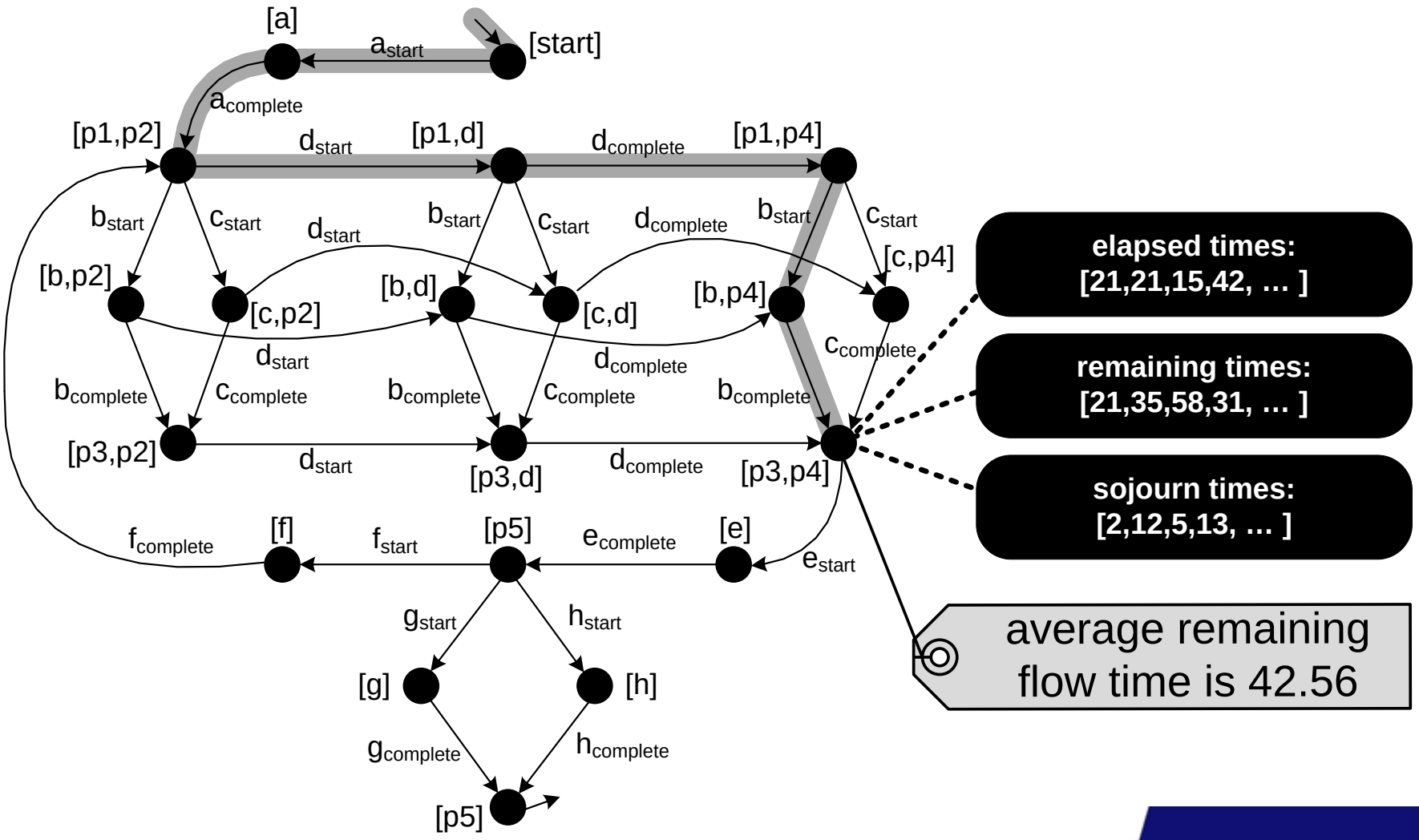
Examples of predictions

- the predicted remaining flow time is 14 days;
- the predicted probability of meeting the legal deadline is 0.72;
- the predicted total cost of this case is 4500 euro;
- the predicted probability that activity a will occur is 0.34;
- the predicted probability that person r will work on this case is 0.57;
- the predicted probability that a case will be rejected is 0.67; and
- the predicted total service time is 98 minutes.

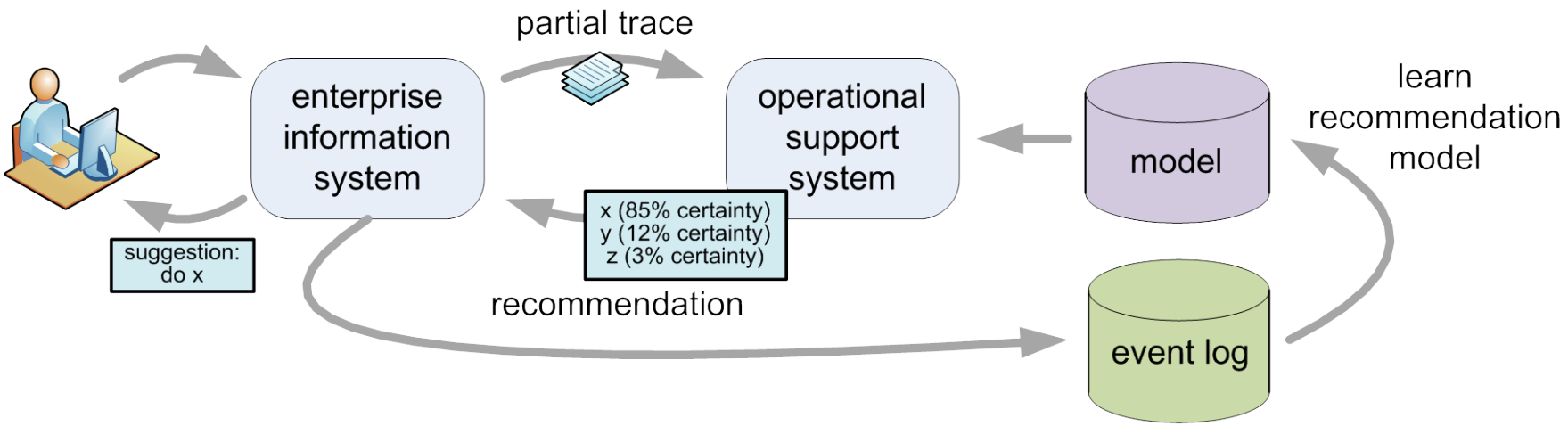
Annotated transition system



Collect results per state



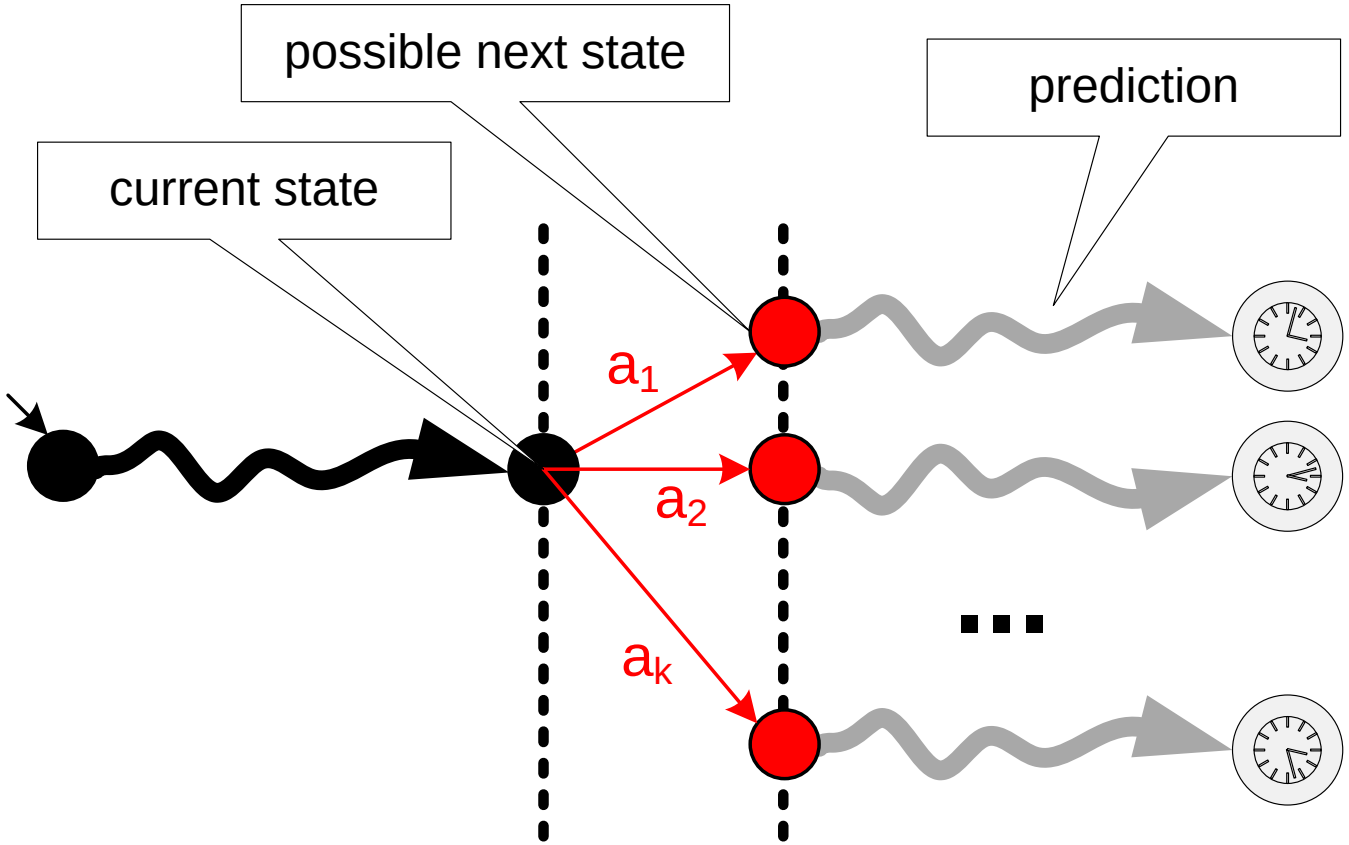
Operational support: Recommend



Recommend

- **Possible recommendations:**
  - **next activity;**
  - **suitable resource; or**
  - **routing decision.**
- **A recommendation is always given with respect to a specific goal.**
- **Examples of goals are:**
  - **minimize the remaining flow time;**
  - **minimize the total costs;**
  - **maximize the fraction of cases handled within 4 weeks;**
  - **maximize the fraction of cases that is accepted; and**
  - **minimize resource usage.**

Relation between prediction and recommendation



Process mining spectrum