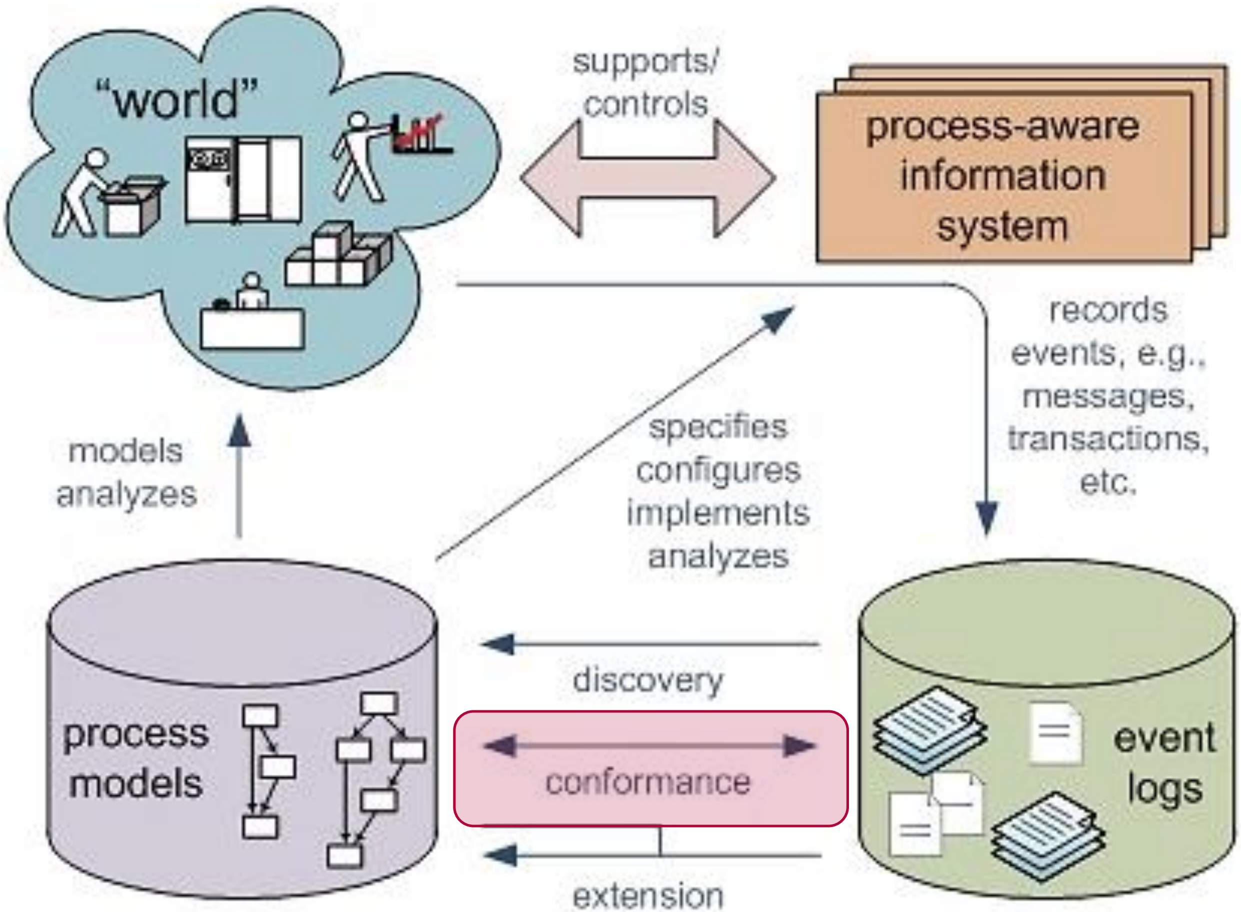
Advanced Process Mining

Prof. Dr. Agnes Koschmider

Lecture 4: Conformance Checking

Lecture Overview

The Context

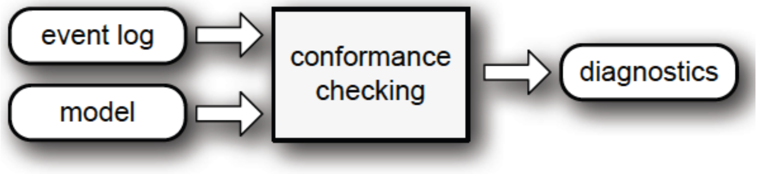

"world"
supports/ controls
process-aware information system
records events, e.g., messages, transactions, etc.
models analyzes
specifies configures implements analyzes
discovery
conformance
extension
process models
event logs

Conformance Checking

Christian-Albrechts-Universität zu Kiel

Conformance checking

- Detect discrepancies between process model and observed information
- Analyse deviations



Perspectives

- Local feedback on deviations at the level of individual traces in the log
- Local feedback on deviations at the level of individual process model parts, e.g., activities
- Global feedback on overall conformance

Motivation

Christian-Albrechts-Universität zu Kiel

Various drivers for conformance assessment

- Corporate governance, risk, compliance, and legislation such as the Sarbanes-Oxley (US), Basel II/III (EU)

- ISO 9001:2008 requires organizations to model their operational processes

- Business alignment: make sure that the information systems and the real business processes are well aligned

Compliance Initiatives

Christian-Albrechts-Universität zu Kiel

Sarbanes Oxley Act of 2002 SOX

- US Federal law enacted on 2002 in response to major corporate scandals (e.g., Enron)
- Aims at increasing trust in public reports on company‘s record
- By following a strict scheme of reporting procedures

Guidelines for Anti Money Laundering

- Money Laundering Control Act (1986), enhanced by USA Patriot Act (2001)
- Aims at preventing money laundering transactions
  - For instance, defined checks must be made before opening a new bank account

How can this be achieved?

- Ensuring existence of specific execution steps in the business processes
- Ensuring order between specific activities in the processes
- If specific steps are missing or they are executed in a wrong order , a compliance violation is detected

Auditing

Auditing is the evaluation of organizations and their processes

- Ascertain the validity and reliability of process information
- Assess whether business processes are executed within boundaries set by managers, governments, and other stakeholders

Process mining to detect fraud, malpractice, risks, and inefficiencies

- Evidence-based, avoiding bias in the retrieval of information
- On-the-fly, while processes are running

How can this be achieved?

- Again, ensuring existence of specific execution steps in the business processes and their execution in a specific order

Setting

So far, fitness measures to evaluate “goodness” of discovered model

- Measure of the fit of data and model
- Independent of any valuation of the data or the model

Now, assuming that the given process model is normative (i.e., the model is “correct”)

- Fitness measures are conformance measures
- Non-fitness is seen as non-conforming process execution
- Non-fitting cases have not been executed as expected

Recall: Replay

Christian-Albrechts-Universität zu Kiel

Given a trace, try to fire transitions in the model accordingly

Not ok:

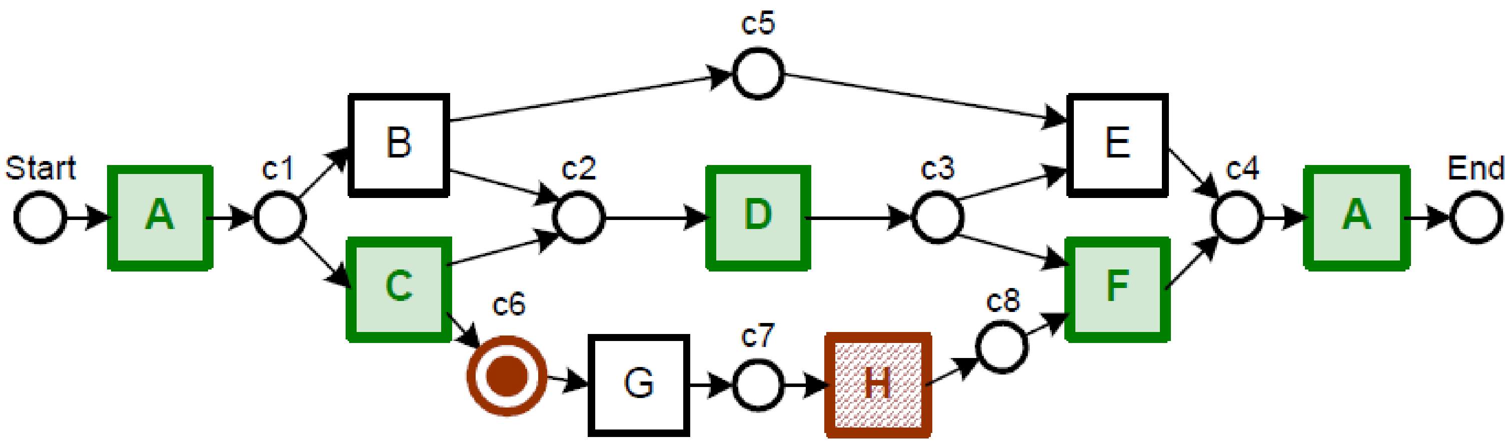
- Transition fires although it is not enabled (missing tokens)
- Tokens remain in the net and are not consumed (remaining tokens)

Replay enables feedback on

- Local non-conformance in a trace
- Local non-conformance in the model
- Global view on conformance

Local Feedback in Trace

Christian-Albrechts-Universität zu Kiel



Per trace:

Transitions that are in line with model and those that are not

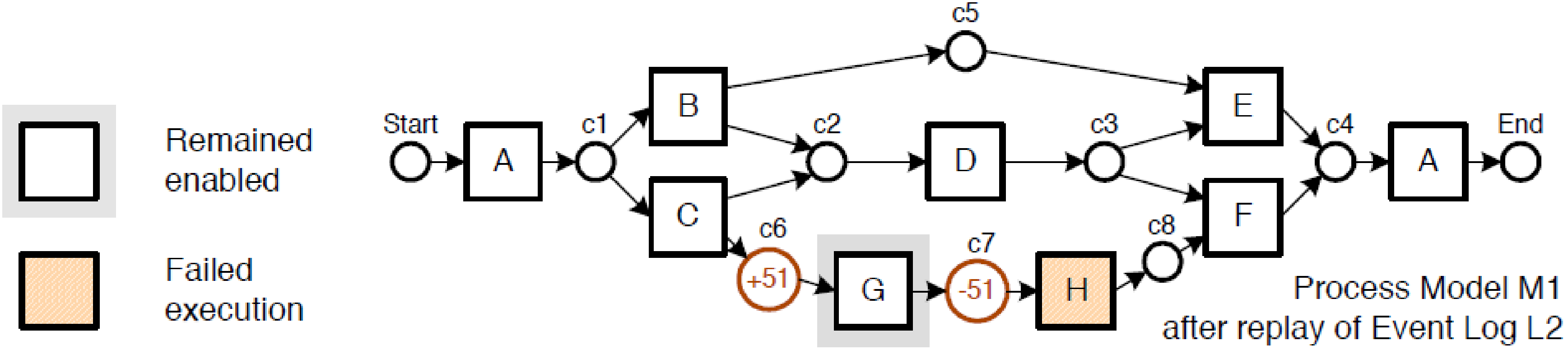
m = 1
r = 1
c = 8
p = 8

No. of Instances	Log Traces
1207	ABDEA
145	ACDGHFA
56	ACGDHFA
23	→ **ACHDFA**
28	ACDHFA

Local Feedback in Model

On the model level:

- Aggregated information about missing and remaining tokens
- Identification of “hotspots” of non-conformance
- Highlights major issues when replaying log and separates some from noise



Global Feedback

C | A | U
Christian-Albrechts-Universität zu Kiel

Overall conformance assessment based on aggregated fitness measure

- Condenses (non-)conformance into a single value
- Yet, hard to interpret in terms of the absolute value

Recall the definition

- k : number of different traces in the log
- n_i : number of occurrences of trace *i* in the log
- m_i : number of missing tokens
- c_i : number of consumed tokens
- r_i : number of tokens that remain in the net
- p_i : number of produced tokens

$$f = \frac{1}{2}\left(1 - \frac{\sum_{i=1}^{k} n_i m_i}{\sum_{i=1}^{k} n_i c_i}\right) + \frac{1}{2}\left(1 - \frac{\sum_{i=1}^{k} n_i r_i}{\sum_{i=1}^{k} n_i p_i}\right)$$

Alignment-based Conformance Checking

Christian-Albrechts-Universität zu Kiel

Assessing the conformance of an event log with a model based on an alignment of activities and events

- Consider the set of activities of the model as a set of symbols
- Then, each execution sequence of the model is a sequence of symbols
- Each trace of the event log is also a sequence of symbols

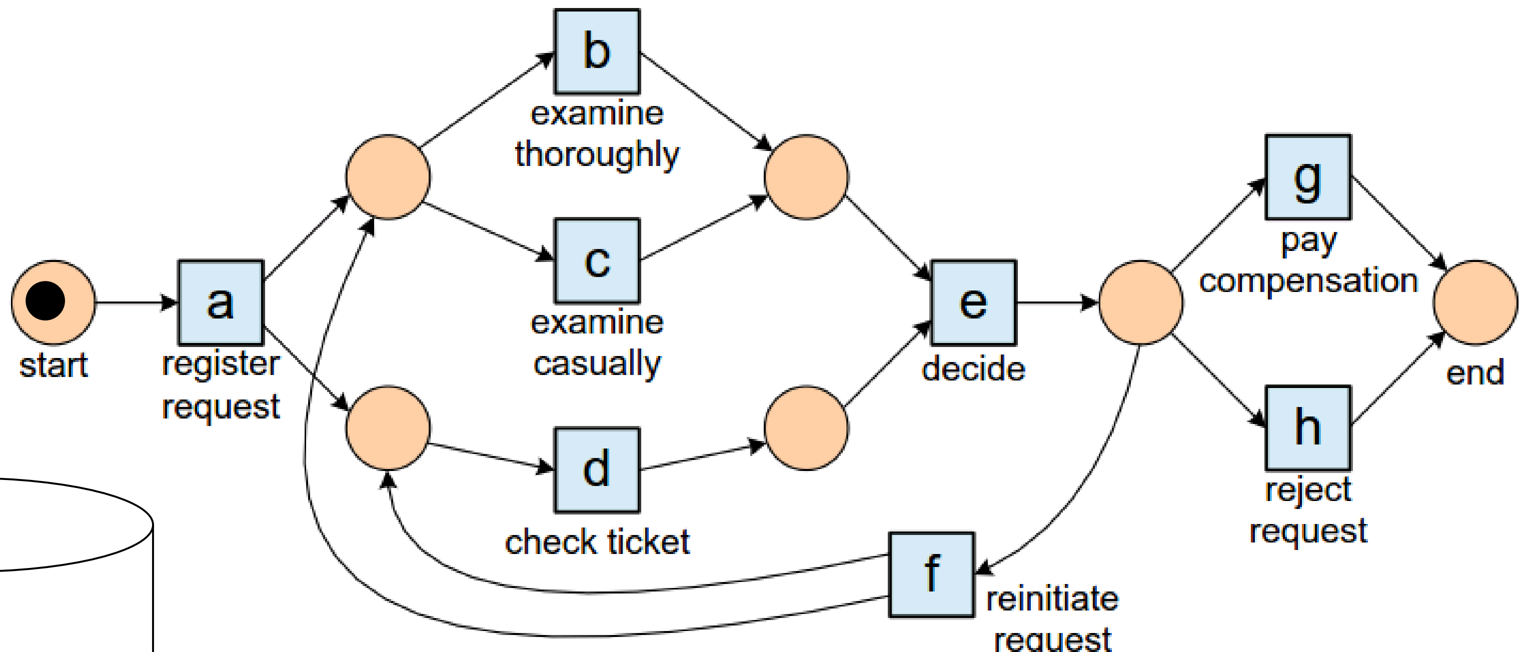
An alignment between two sequences is established by

- Linking pairs of symbols in each sequence
- Such that the order between aligned symbols is preserved

The notion of an alignment allows for quantification of conformance and insights on non-conformance

Example

Model



abfdg

…

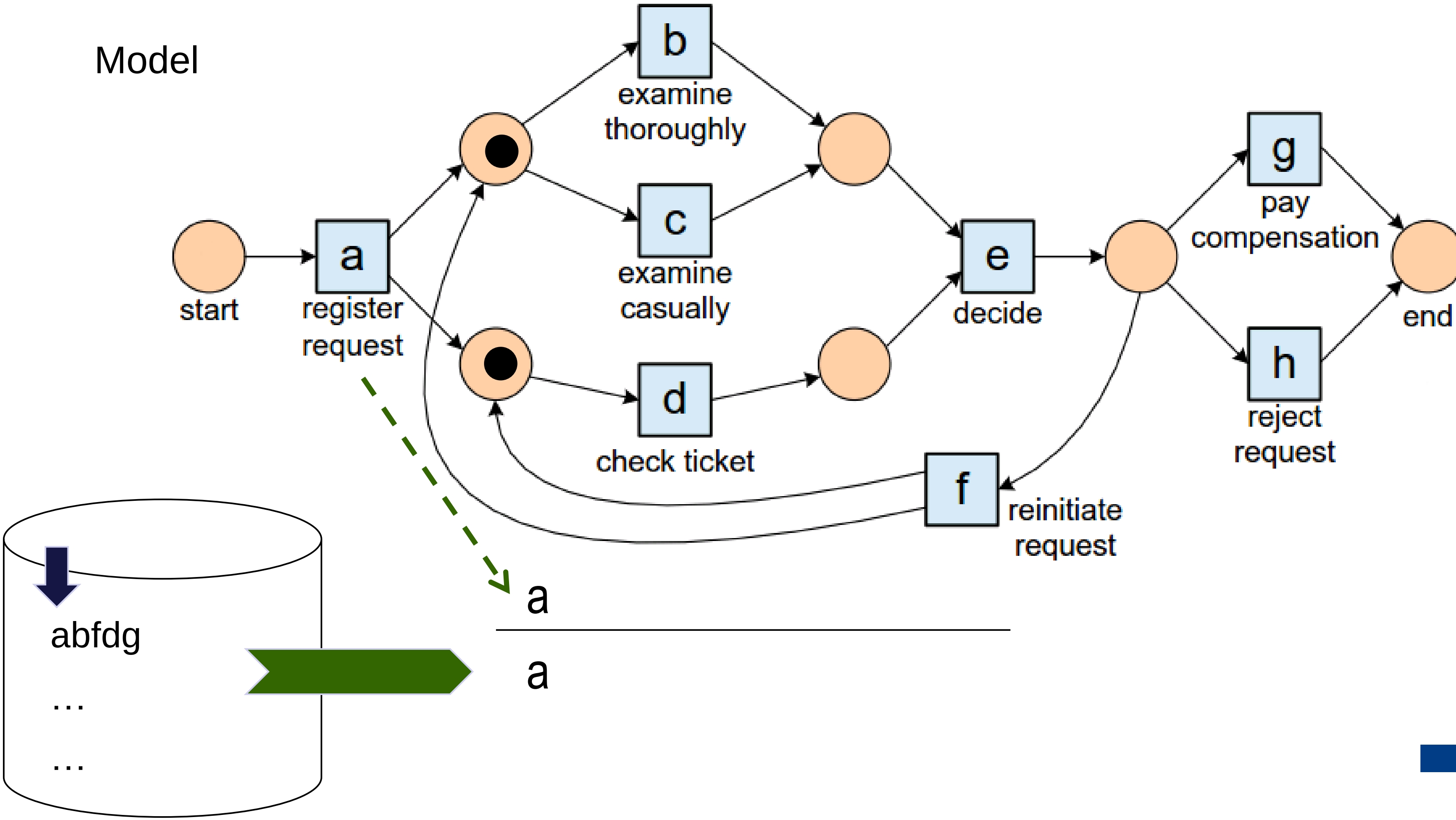
…

Event log

Slides partly due to Boudewijn van Dongen

Example

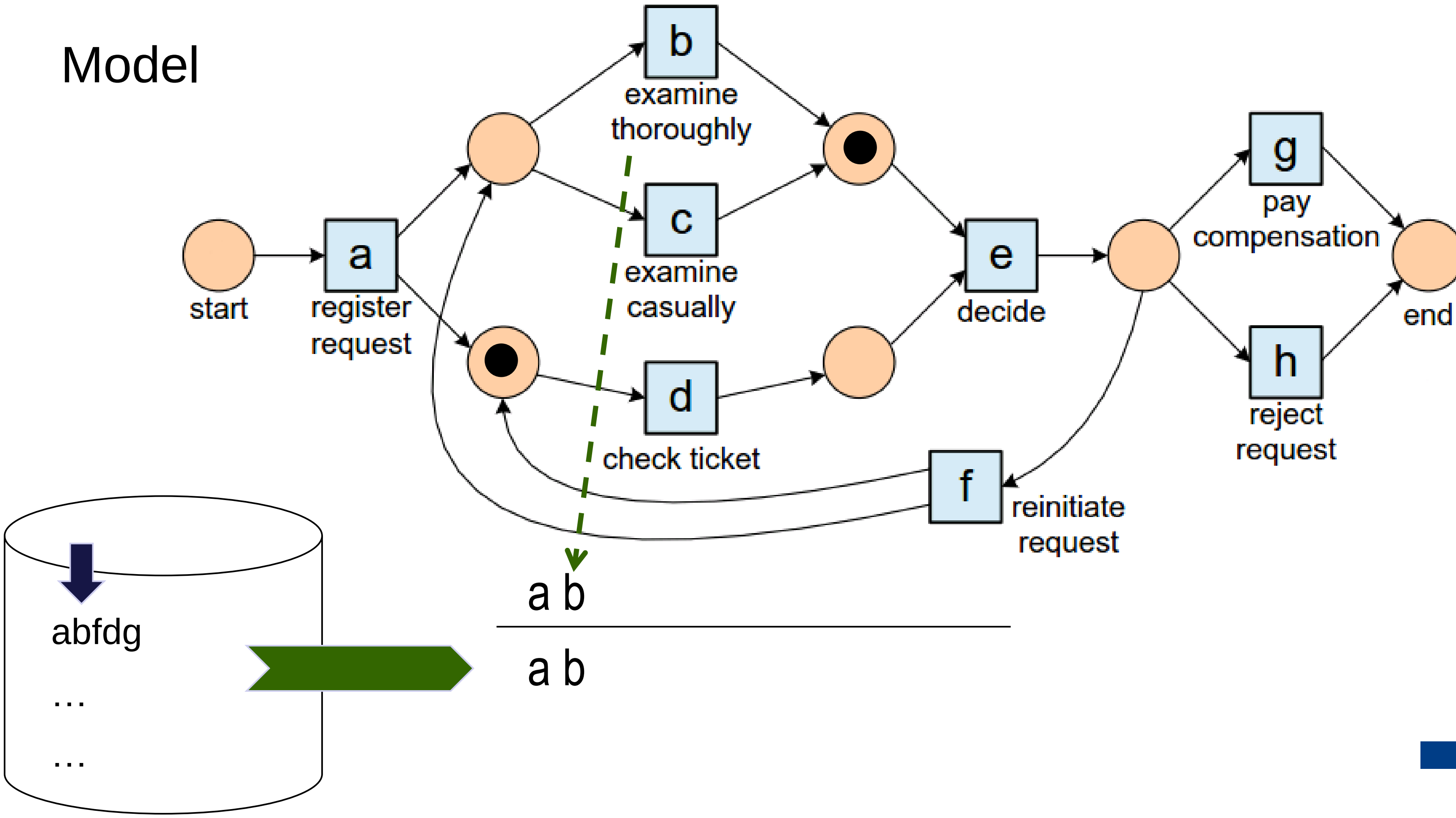
Model



Event log

Alignment

Example

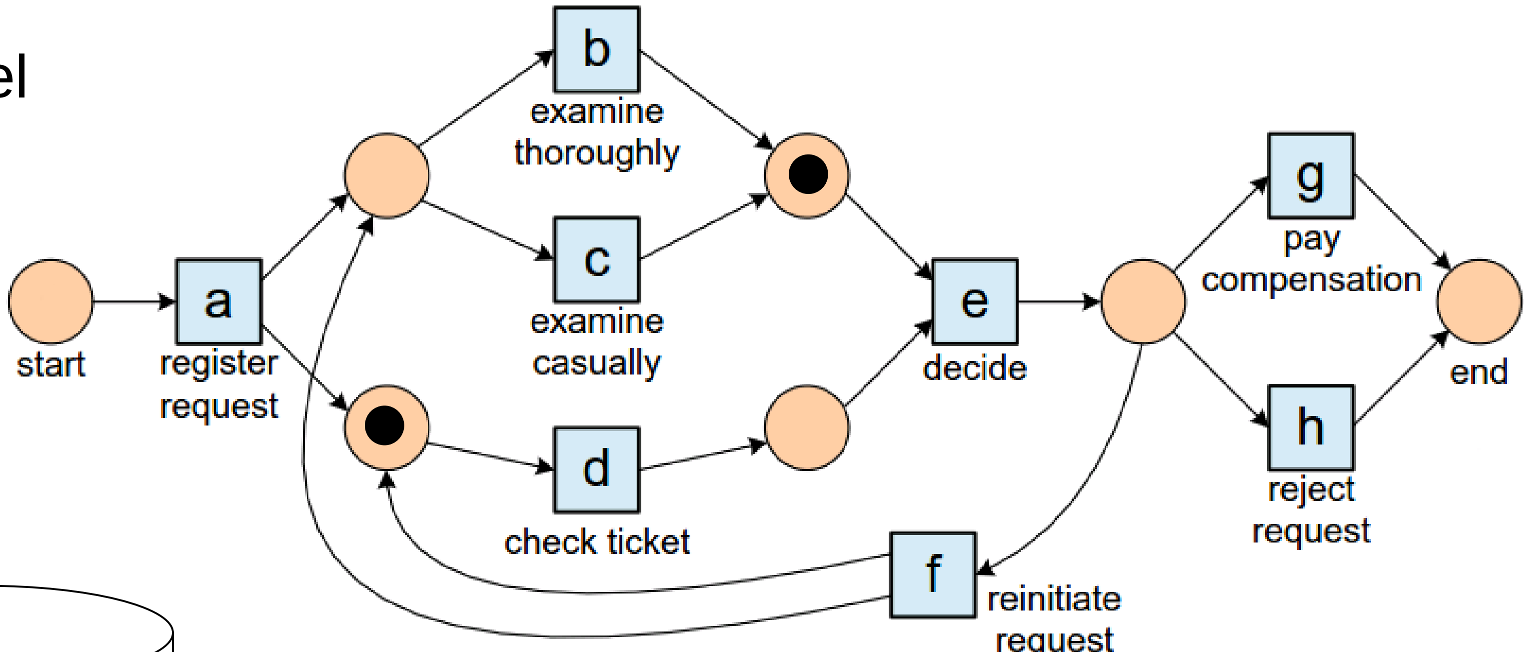


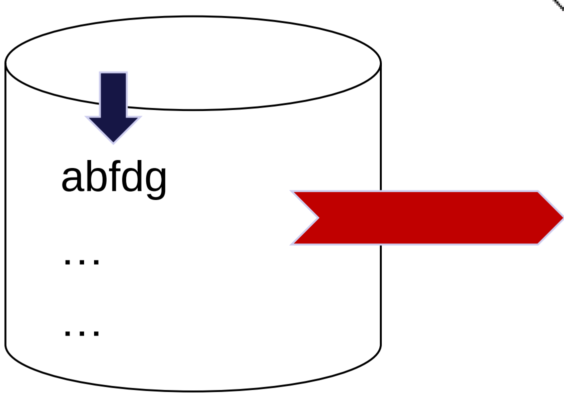
Event log

Alignment

Example

Model





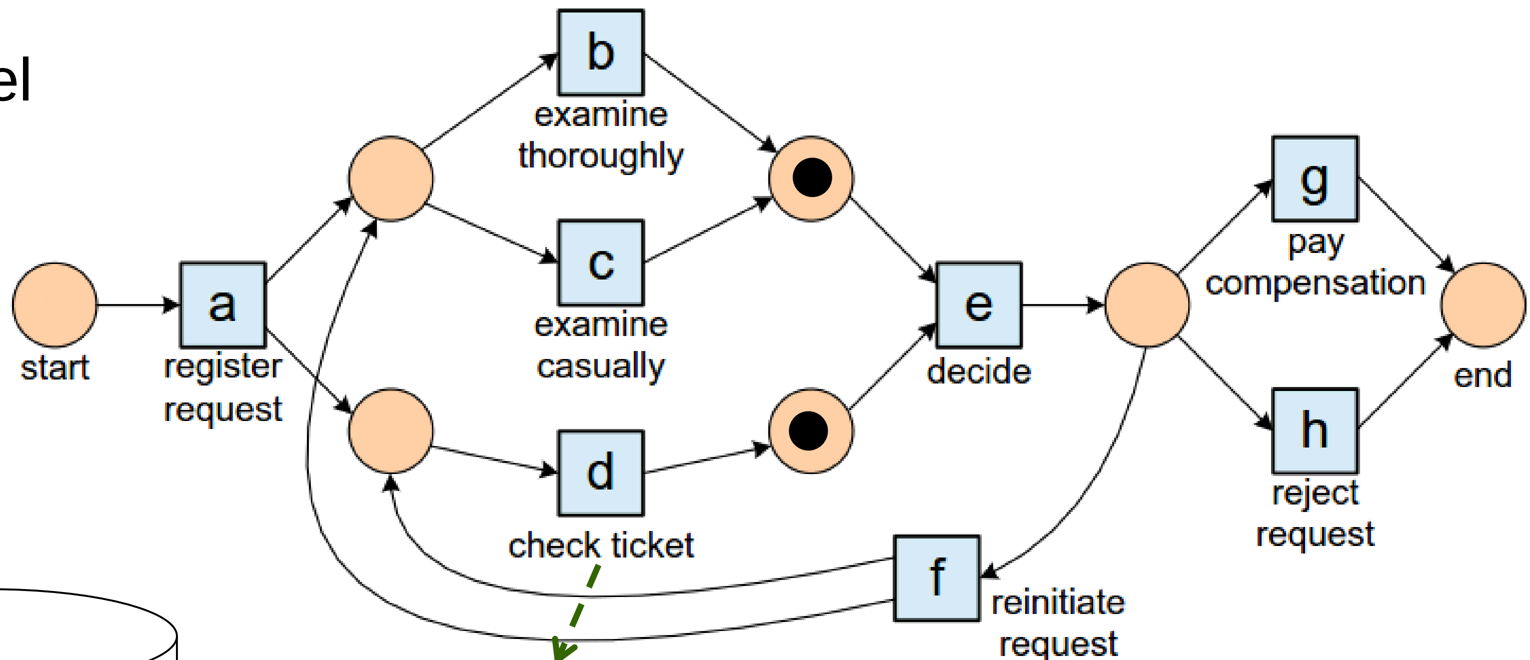
Event log

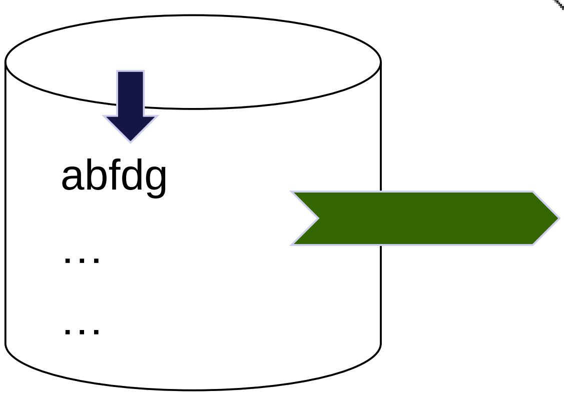
a	b	
a	b	f

Alignment

Example

Model





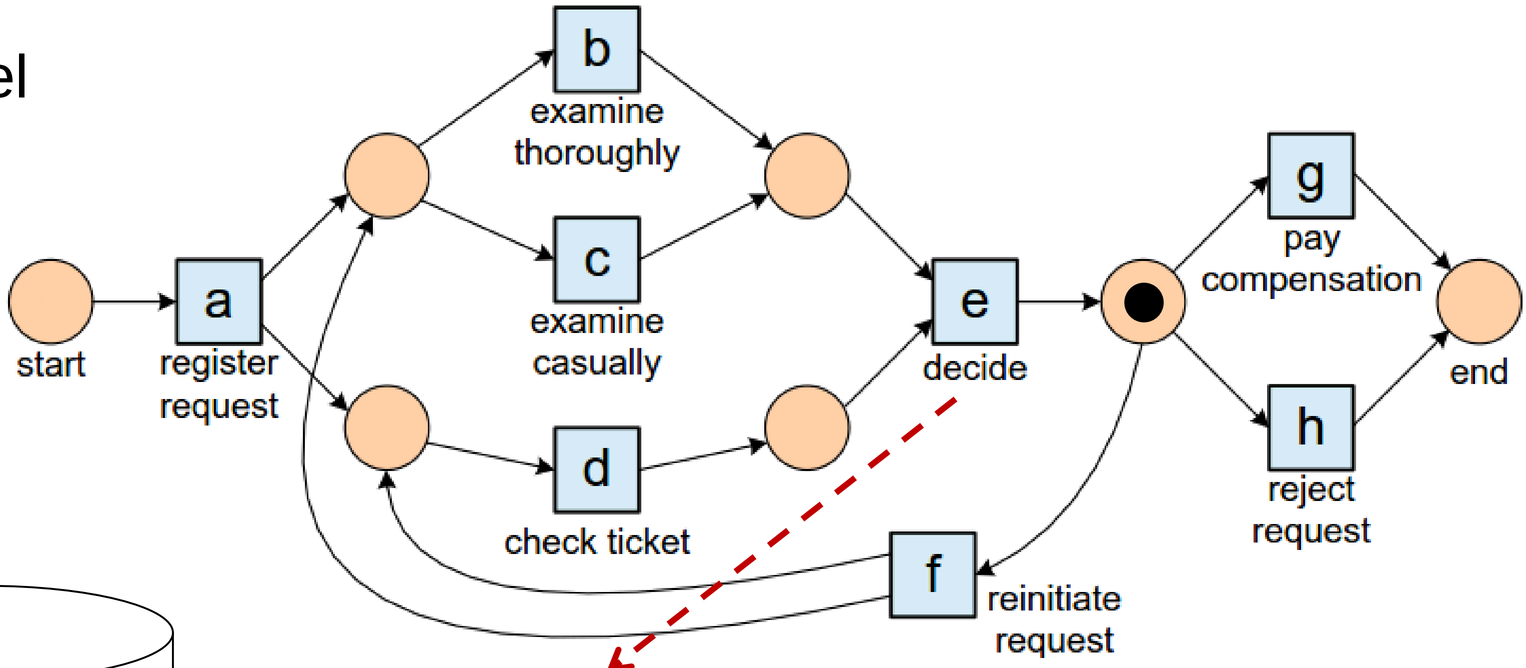
Event log

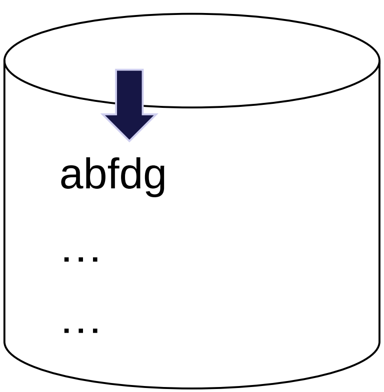
a	b		d
a	b	f	d

Alignment

Example

Model



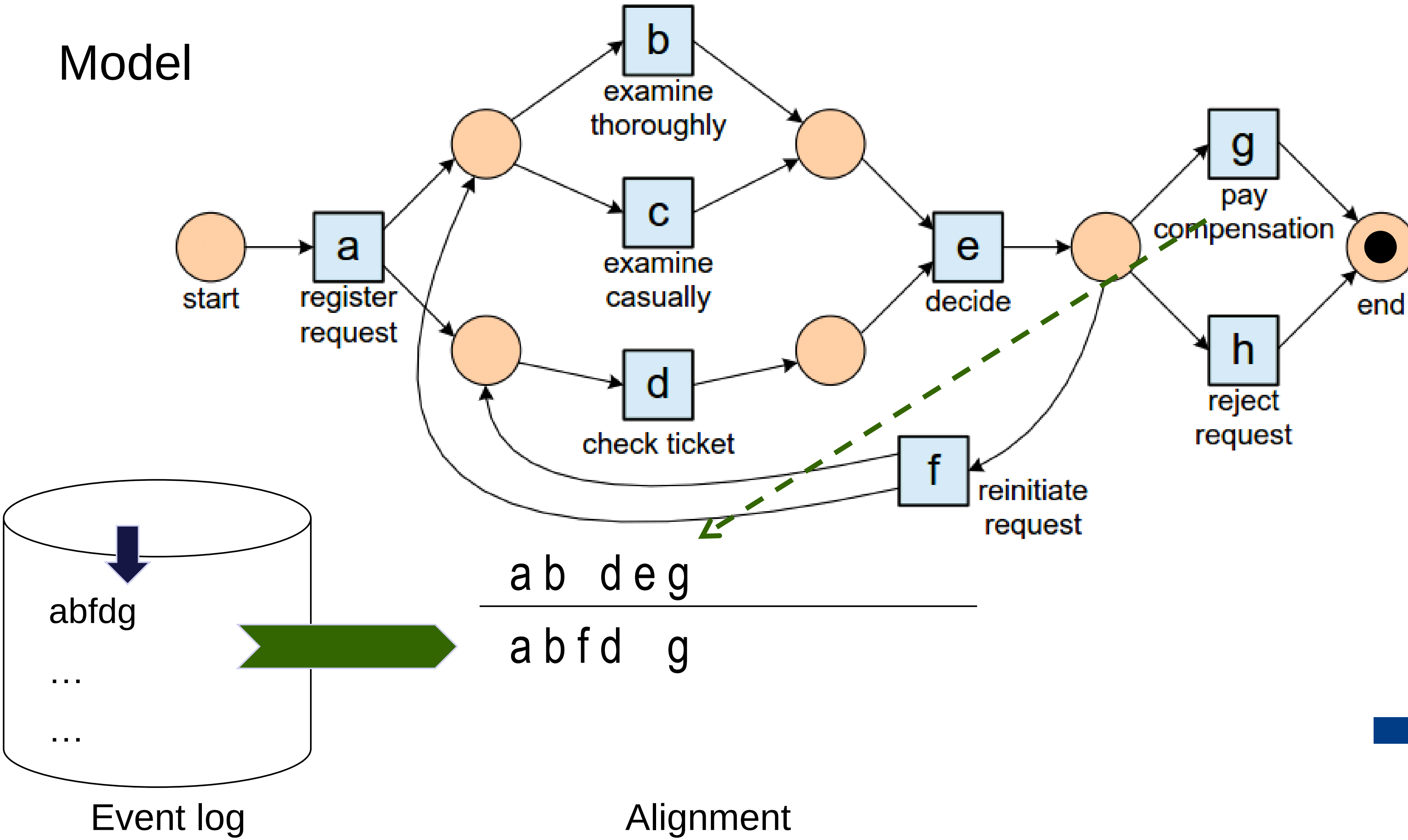


Event log

a	b		d	e
a	b	f	d	

Alignment

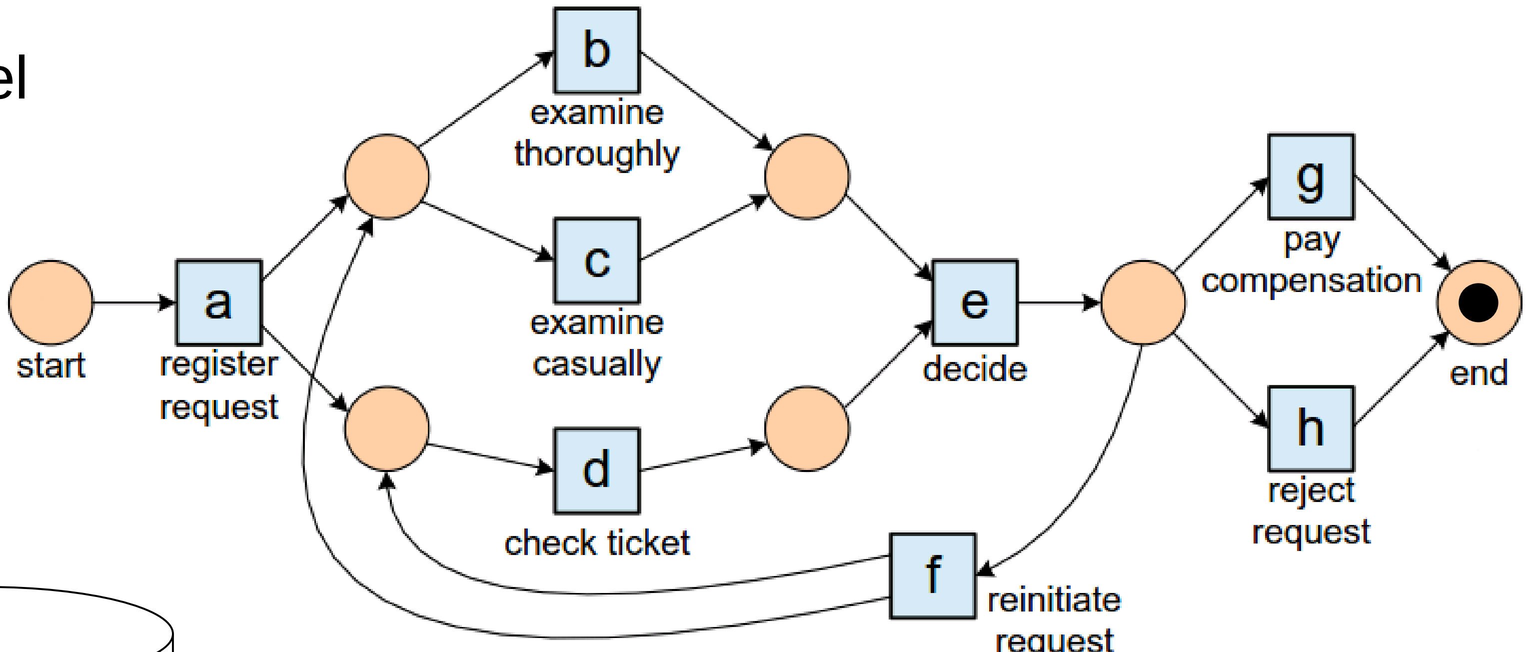
Example



Model
b
examine thoroughly
c
examine casually
d
check ticket
start
a
register request
e
decide
g
pay compensation
h
reject request
end
f
reinitiate request
abfdg
…
…
Event log
a b d e g
a b f d g
Alignment

Example

Model



abfdg
…
…

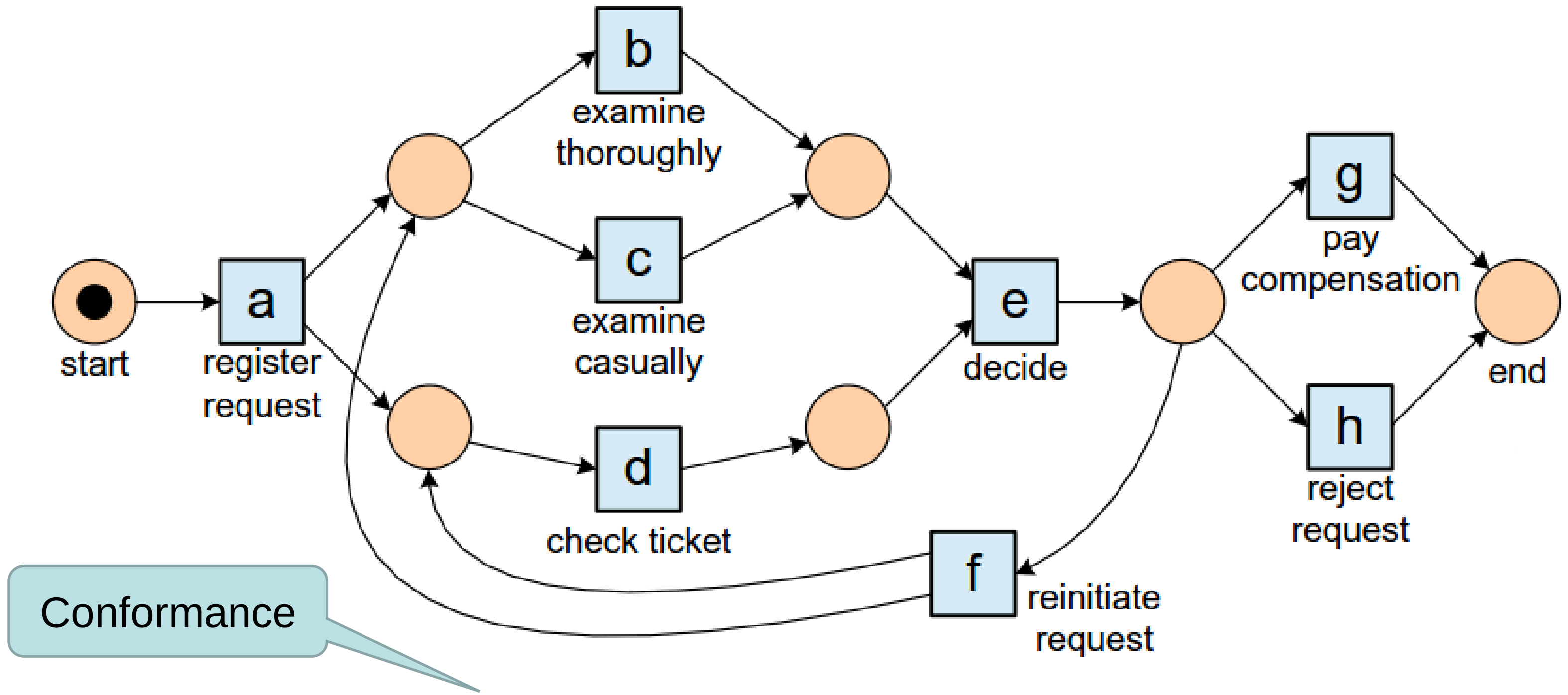
Event log

a	b		d	e	g
a	b	f	d		g

Alignment

Non-Conformance

Idea Cont.



Execution Sequence	a	d	b	e	f	d	c	e	g
Trace	a	d	b	e	f	d	c	e	g

Approach

Christian-Albrechts-Universität zu Kiel

Steps to realise the idea of alignment-based conformance analysis

- Definition of alignments
- Construction of (optimal) alignments
- Conformance measures

Here, focus is on basic approach, various variations have been presented in the literature:

- For different types of process models
- Using different strategies to construct optimal alignments
- Incorprating weighting schemes to fine-tune the importance of activities for conformance checking

Details:
Wil M. P. van der Aalst, Arya Adriansyah, Boudewijn F. van Dongen: Replaying history on process models for conformance checking and performance analysis. Wiley Interdisc. Rew.: Data Mining and Knowledge Discovery (WIDM) 2(2):182-192 (2012)

The Notion of Moves

Alignment is based on "moves" in trace or execution sequence

- Specific symbol $\perp$ to denote "no move"
- Set T_σ as the transitions of trace σ and $T_\sigma^\perp = T_\sigma \cup \{\perp\}$
- Set T_π as the transitions of execution sequence π and $T_\pi^\perp = T_\pi \cup \{\perp\}$

One *step* is a pair $(x, y) \in T_\sigma^\perp \times T_\pi^\perp$ and

- (x, y) is a move in log if $\mathrm{x} \in T_\sigma$ and $y = \perp$
- (x, y) is a move in model if $x = \perp$ and $y \in T_\pi$
- (x, y) is a move in both if $x \in T_\sigma$ and $y \in T_\pi$
- (x, y) is an illegal move in log if $x = \perp$ and $y = \perp$

$T_{\sigma\pi} = \{(x, y) \in T_\sigma^\perp \times T_\pi^\perp \mid x \in T_\sigma \vee y \in T_\pi\}$ is the set of all legal moves

Alignment Definition

An alignment of trace σ and execution sequence π is a sequence of steps $\gamma \in T^*_{\sigma\pi}$, such that

- The projection of γ on its first component, ignoring $\perp$, is σ
- The projection of γ on its second component, ignoring $\perp$, is π

Example:

a	b	$\perp$	d	e	g	$\perp$
a	$\perp$	c	d	e	$\perp$	h

Note: Given a trace and an execution sequence

- There is more than one alignment
- The set of possible alignments is finite

Alignment Examples

$$\gamma_1 = \begin{array}{|c|c|c|c|c|} a & c & d & e & h \\ \hline a & c & d & e & h \end{array}$$

"Ideal" alignment

$$\gamma_2 = \begin{array}{|c|c|c|c|c|c|c|} a & b & \perp & d & e & g & \perp \\ \hline a & \perp & c & d & e & \perp & h \end{array}$$

"Imperfect" alignment

$$\gamma_3 = \begin{array}{|c|c|c|c|c|c|c|c|} a & b & \perp & d & e & \perp & \perp & g \\ \hline \perp & a & c & d & \perp & e & h & \perp \end{array}$$

"Non-optimal" alignment

$$\gamma_4 = \begin{array}{|c|c|c|c|c|c|c|} a & \perp & b & d & e & g & \perp \\ \hline a & \perp & c & d & e & \perp & h \end{array}$$

Not an alignment

Cost of Moves

Introduce cost for legal moves

- Function δ: $T_{\sigma\pi} \rightarrow \mathbb{N}$ assigns cost to each move
- Typically, $\delta(x, x) = 0$

Standard cost function, for $x \in T_\sigma$ and $y \in T_\pi$:

- $\delta(x, \perp) = 1$ (move in log)
- $\delta(\perp, y) = 1$ (move in model)
- $\delta(x, y) = 0$ if $x = y$ (equal move in both)
- $\delta(x, y) = \infty$ if $x \neq y$ (different move in both)

Various further cost functions possible

- Take into account importance of activities (skip payment vs. skip logging)
- Consider similarity of activities (preliminary check vs. partial check)

Examples Again

$$\gamma_1 = \begin{array}{|c|c|c|c|c|} a & c & d & e & h \\ \hline a & c & d & e & h \end{array}$$

Cost: 0

$$\gamma_2 = \begin{array}{|c|c|c|c|c|c|c|} a & b & \perp & d & e & g & \perp \\ \hline a & \perp & c & d & e & \perp & h \end{array}$$

Cost: 4

$$\gamma_3 = \begin{array}{|c|c|c|c|c|c|c|c|} a & b & \perp & d & e & \perp & \perp & g \\ \hline \perp & a & c & d & \perp & e & h & \perp \end{array}$$

Cost: ∞

Optimal Alignment

So far, cost of *some* alignment of trace with respect to *some* execution sequence

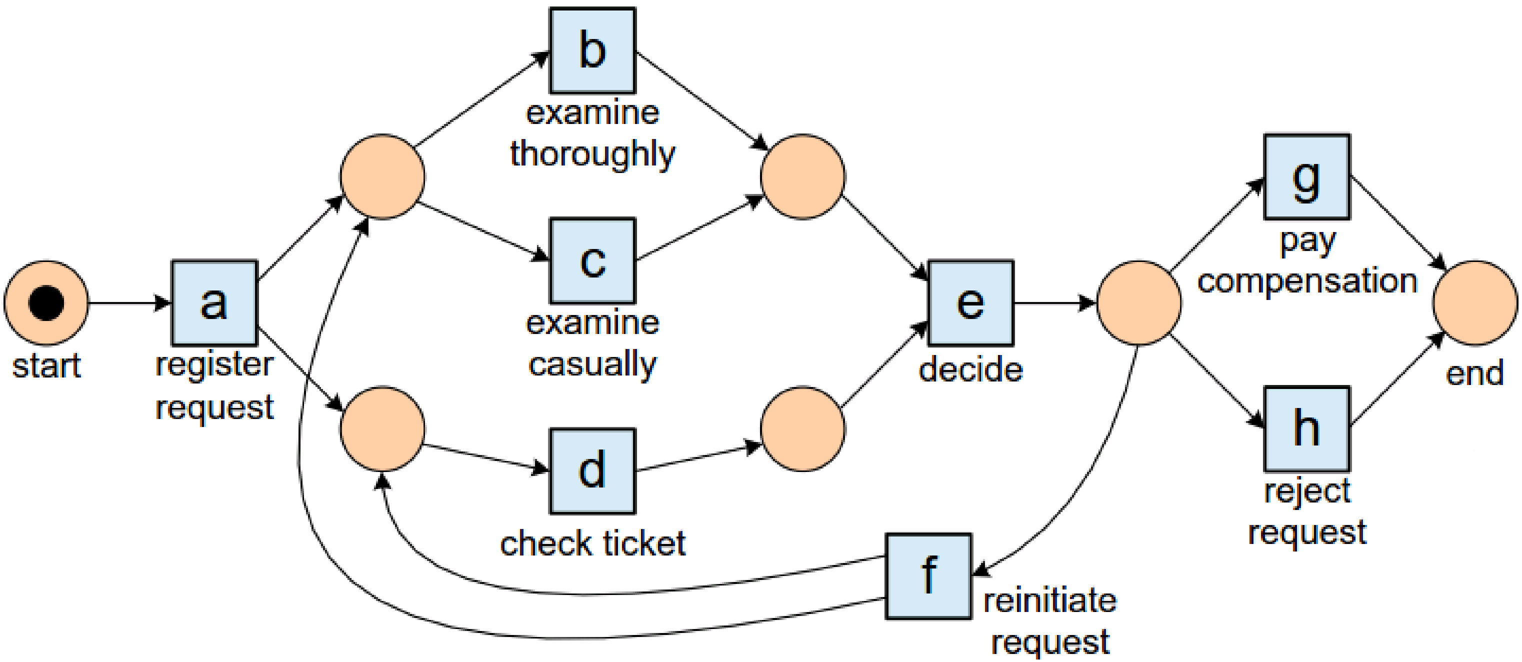
- Optimal alignment of trace σ and execution sequence π: alignment with minimal cost
- Optimal alignment of trace σ and model, i.e., set of complete execution sequences Π: alignment with minimal cost of all optimal alignments of σ and complete execution sequence $\pi \in \Pi$
- Notation: optimal alignment of σ and execution sequence π is $\gamma^*(\sigma, \pi)$, optimal alignment of σ and execution sequences Π is $\gamma^*(\sigma, \Pi)$

Optimal alignment of trace σ and model always exists

- Maybe the trivial alignment that first only moves in trace and then in shortest complete execution sequence, or vice versa
- Not unique, multiple alignments may show minimal cost

Finding the optimal alignment is expensive in general

Example



Trace: abefbh

Cost: 3

a	b	d	e	f	b	d	e	h
a	b	$\perp$	e	f	b	$\perp$	$\perp$	h

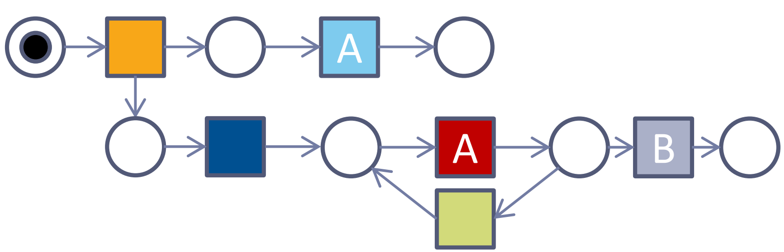
The Problem of Finding Optimal Alignments

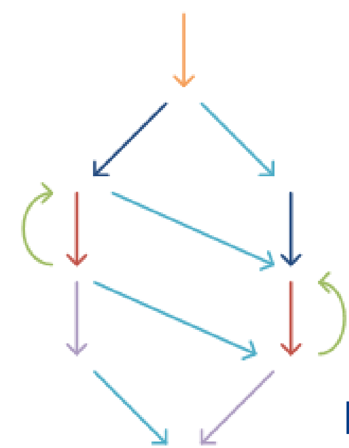
The search space is a “product” of the statespace of the model and the trace

Each node is a combination of a state in the model and the executed events in the trace

Each arc is a move in model, move in log or move in both

Example:





9 states,
13 transitions

Trace: < A, A, B, A >

Chapter 3