Business Process Intelligence Course

〈 Lecture 5 〉

Conformance Checking

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

## 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
Conformance checking

“world”
- business
- processes
- people
- machines
- components
- organizations

models analyzes

software system
- records events, e.g., messages, transactions, etc.
- specifies configures implements analyzes

conformance
- discovery
- enhancement

(event) model

supports/controls

pects/models

(backend)
Language identification in the limit (Mark Gold 1967)

Relation to process mining

sentence $\cong$ trace in event log
language $\cong$ process model
So conformance checking is like spell checking ...

- An activity that should not happen happened
- An activity was executed by the wrong person
- An activity was executed too late
- An activity that should happen did not happen
- Two activities were swapped
need for conformance checking
Using conformance checking

- Local diagnostics
- Global conformance measures
- Local diagnostics

Event log

Process model
Context

- Corporate governance, risk, compliance, and legislation such as the Sarbanes-Oxley (US), Basel II/III (EU), J-SOX (Japan), C-SOX (Canada), 8th EU Directive (EURO-SOX), BilMoG (Germany), MiFID (EU), Law 262/05 (Italy), Code Lippens (Belgium), and Code Tabaksblat (Netherlands).

- 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.
Auditing

- Audits are performed to ascertain the validity and reliability of information about these organizations and associated processes.
- This is done to check whether business processes are executed within certain boundaries set by managers, governments, and other stakeholders.
- Obviously, process mining can help to detect fraud, malpractice, risks, and inefficiencies.
- All events in a business process can be evaluated and this can also be done while the process is still running.
Deviations?

- Is the model or the log "wrong"?
- "Desirable" or "undesirable" deviations?
- "Breaking the glass" may save lives!
Another important use case: Evaluation of process mining algorithms

Model 1 produced by algorithm A
Model 2 produced by algorithm B
Model 3 produced by algorithm C
Model 4 produced by algorithm D
Replay: Connecting events to model elements is essential for process mining

Play-In

- event log
- process model

Play-Out

- process model
- event log

Replay

- event log
- process model
- extended model showing times, frequencies, etc.
- diagnostics
- predictions
- recommendations
Replay

A B C D

start

p1

E

p3

p4

end
Replay can detect problems

Problem! token left behind

Problem! missing token

start

p1

p2

p3

p4

end
Replay is not just useful for conformance checking!

e.g. bottleneck analysis

\[ A^5B^8C^9D^{13} \]
Conformance checking by playing the "token game"
Replaying trace “abeg”

\[ \text{fitness}(\sigma, N) = \frac{1}{2} \left( 1 - \frac{1}{6} \right) + \frac{1}{2} \left( 1 - \frac{1}{6} \right) = 0.83333 \]
Approach

• Use four counters:
  • \( p \) = produced tokens
  • \( c \) = consumed tokens
  • \( m \) = missing tokens
  • \( r \) = remaining tokens

• Invariants:
  • At any time: \( p + m \geq c \geq m \) (also per place)
  • At the end: \( r = p + m - c \) (also per place)

• Special actions:
  • In the beginning a token will be produced for the source place: \( p = 1 \).
  • At the end a token is removed from the sink place (also if not there): \( c' = c + 1 \).
Replaying (1/3) \( \sigma_1 \) on \( N_1 \)

\[ \sigma_1 = \langle a, c, d, e, h \rangle \]
Replaying (2/3)

\[ \sigma_1 = \langle a, c, d, e, h \rangle \]

Graph with transitions and states:

- Start state: p=4, c=2, m=0, r=0
- End state: p=5, c=3, m=0, r=0
Replaying (3/3)

$\sigma_1 = \langle a, c, d, e, h \rangle$

No problems found!
Replaying (1/3)
\(\sigma_3\) on N_2

\(\sigma_3 = (a, d, c, e, h)\)
Replaying (2/3)

\[ \sigma_3 = \langle a, d, c, e, h \rangle \]
Replaying (3/3)

\[ \sigma_3 = \langle a, d, c, e, h \rangle \]

\[
\begin{array}{c}
 p=5 \\
c=4 \\
m=1 \\
r=0 \\
\end{array}
\]

\[
\begin{array}{c}
p=6 \\
c=5 \\
m=1 \\
r=0 \\
\end{array}
\]

\[
\begin{array}{c}
p=6 \\
c=6 \\
m=1 \\
r=1 \\
\end{array}
\]
Problems encountered when replaying $\sigma_3$ on $N_2$

- One missing token (of 6 consumed tokens)
- One remaining token (of 6 produced tokens)

$$\sigma_3 = \langle a, d, c, e, h \rangle$$

$$fitness(\sigma, N) = \frac{1}{2} \left( 1 - \frac{m}{c} \right) + \frac{1}{2} \left( 1 - \frac{r}{p} \right)$$
Computing fitness at trace level

\[
\text{fitness}(\sigma_3, N_2) = \frac{1}{2} \left(1 - \frac{1}{6}\right) + \frac{1}{2} \left(1 - \frac{1}{6}\right) = 0.8333
\]
Computing fitness at the log level

\[
\text{fitness}(L, N) = \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N, \sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N, \sigma}} \right) + \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N, \sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N, \sigma}} \right)
\]

Looks scary, but is just the sums of \(p, c, m,\) and \(r\) over the multiset of traces in the event log …
Compute fitness

<table>
<thead>
<tr>
<th>#</th>
<th>trace</th>
</tr>
</thead>
<tbody>
<tr>
<td>455</td>
<td>acdeh</td>
</tr>
<tr>
<td>191</td>
<td>abdeg</td>
</tr>
<tr>
<td>177</td>
<td>adceh</td>
</tr>
<tr>
<td>144</td>
<td>abdeh</td>
</tr>
<tr>
<td>111</td>
<td>acdeg</td>
</tr>
<tr>
<td>82</td>
<td>adceg</td>
</tr>
<tr>
<td>56</td>
<td>abbeh</td>
</tr>
<tr>
<td>47</td>
<td>acdefdbbeh</td>
</tr>
<tr>
<td>38</td>
<td>abdeg</td>
</tr>
<tr>
<td>33</td>
<td>acdefbdeh</td>
</tr>
<tr>
<td>14</td>
<td>acdefbdeg</td>
</tr>
<tr>
<td>11</td>
<td>acdefdbdeg</td>
</tr>
<tr>
<td>9</td>
<td>acdefcdeh</td>
</tr>
<tr>
<td>8</td>
<td>acdefdbbeh</td>
</tr>
<tr>
<td>5</td>
<td>acdefbdeg</td>
</tr>
<tr>
<td>3</td>
<td>acdefbdefdbbeg</td>
</tr>
<tr>
<td>2</td>
<td>acdefdbbeg</td>
</tr>
<tr>
<td>2</td>
<td>acdefbdefbdeg</td>
</tr>
<tr>
<td>1</td>
<td>acdefbdefbdeh</td>
</tr>
<tr>
<td>1</td>
<td>abefbdefdbbeg</td>
</tr>
<tr>
<td>1</td>
<td>acdefbdefcdefdbbeg</td>
</tr>
</tbody>
</table>

©Wil van der Aalst TU/e (use only with permission & acknowledgements)
Computed fitness

\[
\text{fitness}(L, N) = \frac{1}{2} \left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N, \sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N, \sigma}}\right) + \frac{1}{2} \left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N, \sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N, \sigma}}\right)
\]

- \(\text{fitness}(L_{\text{full}}, N_1) = 1\)
- \(\text{fitness}(L_{\text{full}}, N_2) = 0.9504\)
- \(\text{fitness}(L_{\text{full}}, N_3) = 0.8797\)
- \(\text{fitness}(L_{\text{full}}, N_4) = 1\)
Diagnostics

\[
(fitness(L_{full}, N_2) = 0.9504)
\]

**Problem**
443 tokens remain in place p2, because \(d\) did not occur although the model expected \(d\) to happen.

**Problem**
443 tokens were missing in place p2 during replay, because \(d\) happened even though this was not possible according to the model.

\[
fitness(L, N) = \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N, \sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N, \sigma}} \right) + \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N, \sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N, \sigma}} \right)
\]
Diagnostics

\[ \text{fitness}(L_{full}, N_3) = 0.8797 \]

**Problem:**
- 430 tokens remain in place \( p_1 \), because \( c \) did not happen while the model expected \( c \) to happen

**Problem:**
- 566 tokens were missing in place \( p_3 \) during replay, because \( e \) happened while this was not possible according to the model

**Problem:**
- 10 tokens were missing in place \( p_1 \) during replay, because \( c \) happened while this was not possible according to the model

**Problem:**
- 146 tokens were missing in place \( p_2 \) during replay, because \( d \) happened while this was not possible according to the model

**Problem:**
- 607 tokens remain in place \( p_5 \), because \( h \) did not happen while the model expected \( h \) to happen

**Problem:**
- 461 of the 1391 cases did not reach place end
Drilling down

- Global conformance measures
- Drill down
- New event log: starting point for process and data mining techniques
- Local diagnostics
- Replay
Not just replay fitness …

- **fitness**: Ability to explain observed behavior
- **precision**: Avoiding underfitting
- **generalization**: Avoiding overfitting
- **simplicity**: Occam’s Razor

**Process Mining**

- **lift**: Thrust
- **drag**: Gravity
examples
Exercise 1:
Compute fitness using missing and remaining tokens

- Consider the event log containing 35 cases.
- What is the fitness?
Let us pick one trace: acd

<table>
<thead>
<tr>
<th>trace</th>
<th>frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>abcd</td>
<td>10</td>
</tr>
<tr>
<td>acbd</td>
<td>10</td>
</tr>
<tr>
<td>aed</td>
<td>10</td>
</tr>
<tr>
<td>abd</td>
<td>2</td>
</tr>
<tr>
<td>acd</td>
<td>1</td>
</tr>
<tr>
<td>ad</td>
<td>1</td>
</tr>
<tr>
<td>abbd</td>
<td>1</td>
</tr>
</tbody>
</table>

\[
\text{fitness}(L, N) = \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N, \sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N, \sigma}} \right) + \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N, \sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N, \sigma}} \right)
\]
Fitness = 0.9658

$$fitness(L, N) = \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N, \sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N, \sigma}} \right) + \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N, \sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N, \sigma}} \right)$$

<table>
<thead>
<tr>
<th>Trace</th>
<th>Frequency</th>
<th>Produced Tokens (p)</th>
<th>Remaining Tokens (r)</th>
<th>Consumed Tokens (c)</th>
<th>Missing Tokens (m)</th>
<th>Produced Tokens (all)</th>
<th>Remaining Tokens (all)</th>
<th>Consumed Tokens (all)</th>
<th>Missing Tokens (all)</th>
</tr>
</thead>
<tbody>
<tr>
<td>abcd</td>
<td>10</td>
<td>6</td>
<td>0</td>
<td>6</td>
<td>0</td>
<td>60</td>
<td>0</td>
<td>60</td>
<td>0</td>
</tr>
<tr>
<td>acbd</td>
<td>10</td>
<td>6</td>
<td>0</td>
<td>6</td>
<td>0</td>
<td>60</td>
<td>0</td>
<td>60</td>
<td>0</td>
</tr>
<tr>
<td>aed</td>
<td>10</td>
<td>6</td>
<td>0</td>
<td>6</td>
<td>0</td>
<td>60</td>
<td>0</td>
<td>60</td>
<td>0</td>
</tr>
<tr>
<td>abd</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>10</td>
<td>2</td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td>acd</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>ad</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>abbd</td>
<td>1</td>
<td>6</td>
<td>2</td>
<td>6</td>
<td>2</td>
<td>6</td>
<td>2</td>
<td>6</td>
<td>2</td>
</tr>
</tbody>
</table>

205  
sum p  
7  
sum r  
205  
sum c  
7  
sum m  
fitness  
0.965853659  

© Wil van der Aalst TU/e (use only with permission & acknowledgements)
ProM 5.2 output
(ProM 6 only supports more advanced conformance checking techniques)

30 of 35 cases are fitting (85%)

fitness of 0.965853

total of 7=2+4+1 remaining tokens

total of 7=1+2+4 missing tokens

fitness of 0.96585363

30 of 35 cases are fitting (85%)
Exercise 2: Compute fitness using missing and remaining tokens

- Consider an event log containing just one case composed of one event, $L = \langle e \rangle$.
- What is the fitness?
Replay $\langle e \rangle$

\[
\text{fitness}(L,N) = \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}} \right) + \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}} \right)
\]

\[p = 3\]
\[c = 3\]
\[m = 3\]
\[r = 3\]

fitness = 0!
Result in ProM

fitness = 0

\[
\text{fitness}(L,N) = \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}} \right) + \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}} \right)
\]
Exercise 3:
Compute fitness using missing and remaining tokens

Consider the event log containing 33 cases.
What is the fitness?
Fitness = 0.895705521

<table>
<thead>
<tr>
<th>trace</th>
<th>frequency</th>
<th>produced tokens (p)</th>
<th>remaining tokens (r)</th>
<th>consumed tokens (c)</th>
<th>missing tokens (m)</th>
<th>produced tokens (all)</th>
<th>remaining tokens (all)</th>
<th>consumed tokens (all)</th>
<th>missing tokens (all)</th>
</tr>
</thead>
<tbody>
<tr>
<td>acd</td>
<td>10</td>
<td>5</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>50</td>
<td>0</td>
</tr>
<tr>
<td>bce</td>
<td>10</td>
<td>5</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>50</td>
<td>0</td>
</tr>
<tr>
<td>ace</td>
<td>5</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>25</td>
<td>5</td>
<td>25</td>
<td>5</td>
</tr>
<tr>
<td>bcd</td>
<td>5</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>25</td>
<td>5</td>
<td>25</td>
<td>5</td>
</tr>
<tr>
<td>dca</td>
<td>1</td>
<td>5</td>
<td>3</td>
<td>5</td>
<td>3</td>
<td>5</td>
<td>3</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>abd</td>
<td>1</td>
<td>6</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>6</td>
<td>3</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>d</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

\[
\text{fitness}(L,N) = \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}} \right) + \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}} \right)
\]

<table>
<thead>
<tr>
<th>sum p</th>
<th>sum r</th>
<th>sum c</th>
<th>sum m</th>
</tr>
</thead>
<tbody>
<tr>
<td>163</td>
<td>17</td>
<td>163</td>
<td>17</td>
</tr>
</tbody>
</table>

fitness = 0.895705521
ProM 5.2 output

Fitness of 0.8957

Total of 1 = 1 + 3 + 6 + 6 + 1 remaining tokens

Total of 17 = 1 + 1 + 7 + 5 + 3 missing tokens
Exercise 4:
Compute fitness using missing and remaining tokens

- Consider the event log containing 20 cases.
- What is the fitness?

<table>
<thead>
<tr>
<th>trace</th>
<th>frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>abefcdd</td>
<td>10</td>
</tr>
<tr>
<td>abbeffccd</td>
<td>10</td>
</tr>
</tbody>
</table>
**Fitness = 0.8**

<table>
<thead>
<tr>
<th>trace</th>
<th>frequency</th>
<th>produced tokens (p)</th>
<th>remaining tokens (r)</th>
<th>consumed tokens (c)</th>
<th>missing tokens (m)</th>
<th>produced tokens (all)</th>
<th>remaining tokens (all)</th>
<th>consumed tokens (all)</th>
<th>missing tokens (all)</th>
</tr>
</thead>
<tbody>
<tr>
<td>abefcd</td>
<td>10</td>
<td>9</td>
<td>2</td>
<td>9</td>
<td>2</td>
<td>90</td>
<td>20</td>
<td>90</td>
<td>20</td>
</tr>
<tr>
<td>abbefccd</td>
<td>10</td>
<td>11</td>
<td>2</td>
<td>11</td>
<td>2</td>
<td>110</td>
<td>20</td>
<td>110</td>
<td>20</td>
</tr>
</tbody>
</table>

\[
\text{fitness}(L,N) = \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}} \right) + \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}} \right)
\]
ProM 5.2 output

- total of 40 = 10 + 10 + 20 remaining tokens
- fitness of 0.8

- total of 40 = 20 + 10 + 10 missing tokens
comparing footprints
## Footprints

<table>
<thead>
<tr>
<th>#</th>
<th>trace</th>
</tr>
</thead>
<tbody>
<tr>
<td>455</td>
<td>acdeh</td>
</tr>
<tr>
<td>191</td>
<td>abdeg</td>
</tr>
<tr>
<td>177</td>
<td>adceh</td>
</tr>
<tr>
<td>144</td>
<td>abdeh</td>
</tr>
<tr>
<td>111</td>
<td>acdeg</td>
</tr>
<tr>
<td>82</td>
<td>acdefg</td>
</tr>
<tr>
<td>56</td>
<td>abdeh</td>
</tr>
<tr>
<td>47</td>
<td>acdefdbeh</td>
</tr>
<tr>
<td>38</td>
<td>acdefbeg</td>
</tr>
<tr>
<td>33</td>
<td>acdefbdeh</td>
</tr>
<tr>
<td>14</td>
<td>acdefbeg</td>
</tr>
<tr>
<td>11</td>
<td>acdefdbeg</td>
</tr>
<tr>
<td>9</td>
<td>acdefcdeh</td>
</tr>
<tr>
<td>8</td>
<td>acdefdbeh</td>
</tr>
<tr>
<td>5</td>
<td>acdefdbeg</td>
</tr>
<tr>
<td>3</td>
<td>acdefbdeh</td>
</tr>
<tr>
<td>2</td>
<td>acdefdbeg</td>
</tr>
<tr>
<td>2</td>
<td>acdefbdefdbeg</td>
</tr>
<tr>
<td>1</td>
<td>acdefbdefbdeh</td>
</tr>
<tr>
<td>1</td>
<td>adbfbdefdbeg</td>
</tr>
<tr>
<td>1</td>
<td>acdefdbefbdeh</td>
</tr>
<tr>
<td>1</td>
<td>acdefdbefcdefdbeg</td>
</tr>
</tbody>
</table>

### Diagrams

- **N1**
  - Start: register request (a)
  - Examine thoroughly (b)
  - Examine casually (c)
  - Check ticket (d)
  - Decide (e)
  - Pay compensation (g)
  - Reject request (h)

- **N2**
  - Start: register request (a)
  - Examine thoroughly (b)
  - Examine casually (c)
  - Check ticket (d)
  - Decide (e)
  - Pay compensation (g)
  - Reject request (h)

- **N3**
  - Start: register request (a)
  - Examine casually (c)
  - Check ticket (d)
  - Decide (e)
  - Reject request (h)

- **N4**
  - Start: register request (a)
  - Examine thoroughly (b)
  - Examine casually (c)
  - Decide (e)
  - Reinitiate request (f)
  - Pay compensation (g)
  - Reject request (h)
## Comparing footprints

### $L_{full}$ and $N_1$

<table>
<thead>
<tr>
<th></th>
<th>$a$</th>
<th>$b$</th>
<th>$c$</th>
<th>$d$</th>
<th>$e$</th>
<th>$f$</th>
<th>$g$</th>
<th>$h$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a$</td>
<td>#</td>
<td>→</td>
<td>→</td>
<td>→</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>$b$</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>→</td>
<td>←</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>$c$</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>→</td>
<td>←</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>$d$</td>
<td>←</td>
<td>→</td>
<td>→</td>
<td>→</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>$e$</td>
<td>#</td>
<td>←</td>
<td>←</td>
<td>←</td>
<td>#</td>
<td>→</td>
<td>→</td>
<td>→</td>
</tr>
<tr>
<td>$f$</td>
<td>#</td>
<td>→</td>
<td>→</td>
<td>→</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>$g$</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>$h$</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>#</td>
</tr>
</tbody>
</table>

### Diagram $N_1$

- **Start**: Register request
- **End**: Pay compensation
- **Activities**:
  - Examine request thoroughly
  - Examine request casually
  - Decide to pay compensation
  - Reject request
  - Reinitiate request

---

© Wil van der Aalst TU/e (use only with permission & acknowledgements)
\[ N_1 \]

- **a**: start register request
- **b**: examine thoroughly
- **c**: examine casually
- **d**: check ticket
- **e**: decide
- **f**: reinitiate request
- **g**: pay compensation
- **h**: reject request

**N_2**

\[ L_{full} \text{ and } N_1 \]

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
<th>h</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>#</td>
<td>→</td>
<td>→</td>
<td>→</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>b</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td></td>
<td></td>
<td></td>
<td>→</td>
<td>←</td>
</tr>
<tr>
<td>c</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td></td>
<td></td>
<td></td>
<td>→</td>
<td>←</td>
</tr>
<tr>
<td>d</td>
<td>←</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>#</td>
</tr>
<tr>
<td>e</td>
<td>#</td>
<td>←</td>
<td>←</td>
<td>←</td>
<td>#</td>
<td>→</td>
<td>→</td>
<td>→</td>
</tr>
<tr>
<td>f</td>
<td>#</td>
<td>→</td>
<td>→</td>
<td>→</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>g</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>h</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td></td>
<td>a</td>
<td>b</td>
<td>c</td>
<td>d</td>
<td>e</td>
<td>f</td>
<td>g</td>
<td>h</td>
</tr>
<tr>
<td>---</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>a</td>
<td>#</td>
<td>→</td>
<td>→</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>b</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>→</td>
<td>#</td>
<td>←</td>
<td>#</td>
</tr>
<tr>
<td>c</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>d</td>
<td>#</td>
<td>←</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>e</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>→</td>
<td>→</td>
</tr>
<tr>
<td>f</td>
<td>#</td>
<td>→</td>
<td>→</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>←</td>
<td>#</td>
</tr>
<tr>
<td>g</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>←</td>
<td>#</td>
</tr>
<tr>
<td>h</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>←</td>
<td>#</td>
</tr>
</tbody>
</table>

**N2**

- **a**: start register request
- **b**: examine thoroughly
- **c**: examine casually
- **d**: check ticket
- **e**: decide
- **f**: pay compensation
- **g**: reject request
- **h**: reinitiate request

End **N1**

- **p1**: register request
- **p2**: examine casually
- **p3**: check ticket
- **p4**: pay compensation
- **p5**: reject request
<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
<th>h</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>#</td>
<td>→</td>
<td>→</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>b</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>c</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>←</td>
<td>→</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>d</td>
<td>#</td>
<td>←</td>
<td>←</td>
<td>#</td>
<td>←</td>
<td>→</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>e</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>f</td>
<td>#</td>
<td>→</td>
<td>→</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>g</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td>#</td>
</tr>
<tr>
<td>h</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>←</td>
<td>#</td>
<td>#</td>
<td>#</td>
</tr>
</tbody>
</table>

\[ N_2 \]

\[ L_{full} \text{ and } N_1 \]
## Differences quantified

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
<th>h</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>c</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>d</td>
<td>←: #</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>e</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>←: #</td>
</tr>
<tr>
<td>f</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>←: #</td>
<td></td>
<td></td>
</tr>
<tr>
<td>g</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>h</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\(x:y\) where \(x\) is in log or \(N_1\) and \(y\) in \(N_2\)

\[
1 - \frac{12}{64} = 0.8125
\]
## Diagnostics

(x:y where x is in log or N_1 and y in N_2)

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
<th>h</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>\rightarrow: #</td>
</tr>
<tr>
<td>b</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>\parallel: \rightarrow  \rightarrow: #</td>
</tr>
<tr>
<td>c</td>
<td>\leftarrow: #</td>
<td>\parallel: \leftarrow</td>
<td>\parallel: \leftarrow</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>\leftarrow: #</td>
</tr>
<tr>
<td>d</td>
<td></td>
<td></td>
<td></td>
<td>\leftarrow: #</td>
<td>\leftarrow: #</td>
<td>\leftarrow: #</td>
<td></td>
<td>\rightarrow: #</td>
</tr>
</tbody>
</table>

### N_1

- **a** (start register request)
- **b** (examine thoroughly)
- **c** (examine casually)
- **d** (check ticket)
- **e** (decide)
- **f** (pay compensation)
- **g** (reject request)
- **h** (reinitiate request)

### N_2

- **a** (start register request)
- **b** (examine thoroughly)
- **c** (examine casually)
- **d** (check ticket)
- **e** (decide)
- **f** (reject request)
- **g** (pay compensation)
- **h** (reinitiate request)

**L_{full} and N_1**

**N_2**
conformance checking based on alignments
Connecting event log and model

- Very important!
- Model may be discovered or hand-made.
- Connected during replay.
- Starting point for other types of process mining!
From “playing the token game” to optimal alignments …

191 times “abde...”

<table>
<thead>
<tr>
<th>#</th>
<th>trace</th>
</tr>
</thead>
<tbody>
<tr>
<td>455</td>
<td>acdeh</td>
</tr>
<tr>
<td>191</td>
<td>abdeh</td>
</tr>
<tr>
<td>177</td>
<td>adceh</td>
</tr>
<tr>
<td>144</td>
<td>abdeh</td>
</tr>
<tr>
<td>111</td>
<td>acdeg</td>
</tr>
<tr>
<td>82</td>
<td>adceg</td>
</tr>
<tr>
<td>56</td>
<td>abeh</td>
</tr>
<tr>
<td>47</td>
<td>acdefbdeh</td>
</tr>
<tr>
<td>38</td>
<td>adbeg</td>
</tr>
<tr>
<td>33</td>
<td>acdefbdeh</td>
</tr>
<tr>
<td>14</td>
<td>acdefbdeg</td>
</tr>
<tr>
<td>11</td>
<td>acdefdbeg</td>
</tr>
<tr>
<td>9</td>
<td>acdefcdeh</td>
</tr>
<tr>
<td>8</td>
<td>acdefbdeh</td>
</tr>
<tr>
<td>5</td>
<td>acdefbdeg</td>
</tr>
<tr>
<td>3</td>
<td>acdefbdefdbeg</td>
</tr>
<tr>
<td>2</td>
<td>acdefdbeg</td>
</tr>
<tr>
<td>2</td>
<td>acdefbdefdbeg</td>
</tr>
<tr>
<td>1</td>
<td>acdefbdefbdeh</td>
</tr>
<tr>
<td>1</td>
<td>acdefbdefdbeg</td>
</tr>
<tr>
<td>1</td>
<td>acdefbdefdbeg</td>
</tr>
<tr>
<td>1391</td>
<td></td>
</tr>
</tbody>
</table>
Example alignments

abdeg

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>d</th>
<th>e</th>
<th>g</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>b</td>
<td>d</td>
<td>e</td>
<td>g</td>
</tr>
<tr>
<td>a</td>
<td>b</td>
<td>d</td>
<td>e</td>
<td>g</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>d</th>
<th>e</th>
<th>g</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>b</td>
<td>d</td>
<td>e</td>
<td>g</td>
</tr>
<tr>
<td>a</td>
<td>c</td>
<td>d</td>
<td>e</td>
<td>g</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>d</th>
<th>e</th>
<th>g</th>
<th>g</th>
<th>g</th>
<th>g</th>
</tr>
</thead>
<tbody>
<tr>
<td>»</td>
<td>»</td>
<td>»</td>
<td>»</td>
<td>»</td>
<td>»</td>
<td>»</td>
<td>»</td>
</tr>
<tr>
<td>»</td>
<td>»</td>
<td>»</td>
<td>»</td>
<td>»</td>
<td>a</td>
<td>c</td>
<td>d</td>
</tr>
</tbody>
</table>
Moves in an alignment

- **move in log**

- **trace in event log**

- **possible run of model**

- **move in model**

- **move in both**
Moves have costs

- Standard cost function:
  - $c(x, \rightarrow) = 1$
  - $c(\rightarrow, y) = 1$
  - $c(x, y) = 0$, if $x=y$
  - $c(x, y) = \infty$, if $x\neq y$
Optimal alignment (smallest costs)

abdeg

optimal

a  b  d  e  g

0

a  b  d  e  g

2

a  b  d  e  g

10

a  b  d  e  g

©Wil van der Aalst TU/e (use only with permission & acknowledgements)
Non-fitting trace: abefdeg

abefdeg
Any cost structure is possible


<table>
<thead>
<tr>
<th>...</th>
<th>send-letter(John, 2 weeks, $400)</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>send-email(Sue, 3 weeks, $500)</td>
<td>...</td>
</tr>
</tbody>
</table>

- Similar activities (more similarity implies lower costs).
- Resource conformance (done by someone that does not have the specified role).
- Data conformance (path is not possible for this customer).
- Time conformance (missed the legal deadline).
- cf. cost/risk-aware BPM (costs = risk).
Advantages of aligning log and model

- Observed behavior is directly related to modeled behavior.
- Highly flexible (any cost structure).
- Detailed diagnostics.
- After aligning log and model, other quality dimensions can be investigated (separation of concerns).
- Efficiently implemented in ProM.
Alignments are essential!

- conformance checking to diagnose deviations
- squeezing reality into the model to do model-based analysis
process model

event log

synchronous move

move on model only

move on log only

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>a</strong></td>
<td><strong>c</strong></td>
<td><strong>d</strong></td>
<td><strong>f</strong></td>
<td><strong>h</strong></td>
</tr>
<tr>
<td><strong>a</strong></td>
<td><strong>b</strong></td>
<td><strong>d</strong></td>
<td><strong>τ</strong></td>
<td><strong>t</strong></td>
</tr>
</tbody>
</table>
| *t*1 | *t*4 | *t*3 | *t*5 | *t*7 | *t*10
Loops of “W_Completeren aanvraag” and “W_Nabellen offertes” are often performed

“O_DECLINED” and “W_Wijzigen contractgegevens” are often skipped

Many moves on log of “O_CANCELED”, “O_CREATED”, “O_SELECTED”, “O_SENT” occurred with the same frequency value (i.e. 60) before parallel branch

Many moves on log of “W_Afhandelen leads” ( > 2200 times) occurred in the end of traces

Work of Arya Adriansyah and Boudewijn van Dongen (Replay project)
Loops of “W_Completeren aanvraag” and “W_Nabellen offertes” are often performed.

Many moves on log of “O_DECLINED” and “W_Wijzigen contractgegevens” are often skipped.

Many moves on log of “O_CANCELLED”, “O_CREATED”, “O_SELECTED”, “O_SENT” occurred with the same frequency value (i.e. 60) before parallel branch “W_Afhandelen leads” ( > 2200 times) occurred in the end of traces.

Loops of “W_Completeren aanvraag” and “W_Nabellen offertes” are often performed.

Synchronous moves of “Completeren aanvraag” Move on log of “Completeren aanvraag”

Moves on model towards end of traces Move on log of “O_CANCELLED” and “A_CANCELLED”
The average waiting time for the input place of “W_Nabellen offertes+START” is very long (2.83 days) compares to the average waiting time of other places.

“O_ACCEPTED” has average sojourn time of 27.07 minutes, while “A_REGISTERED”, “A_ACTIVATED”, and “A_APPROVED” have average sojourn time of 29.56 minutes.

Activity “W_Wijzigen contractgegevens” is the bottleneck, but it occured rarely (only 4 times).

**Business analyst's toolbox**
traffic jams are projected on map

also conformance problems can be shown (in real-time if needed)
conclusion and outlook
Conformance checking

"world"
people
machines
components
organizations

models
analyzes

business processes

supports/controls

software system

specifies
configures
implements
analyzes

records
events, e.g.,
messages,
transactions,
etc.

(event)
model

(discovery)

(conformance)

enhancement

©Wil van der Aalst TU/e (use only with permission & acknowledgements)