No Knowledge Without Processes

Process Mining as a Tool to Find Out What People and Organizations Really Do

prof.dr.ir. Wil van der Aalst
www.processmining.org
Process discovery
Process Discovery
Conformance checking
Let's play
Play-Out

<table>
<thead>
<tr>
<th>Case</th>
<th>Activity</th>
<th>Timestamp</th>
<th>Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>432</td>
<td>register travel request (a)</td>
<td>18-3-2014 19:15</td>
<td>John</td>
</tr>
<tr>
<td>432</td>
<td>get support from local manager (b)</td>
<td>19-3-2014 6:25</td>
<td>Mary</td>
</tr>
<tr>
<td>432</td>
<td>check budget by finance (d)</td>
<td>19-3-2014 6:30</td>
<td>John</td>
</tr>
<tr>
<td>432</td>
<td>decide (e)</td>
<td>19-3-2014 6:36</td>
<td>Sue</td>
</tr>
<tr>
<td>432</td>
<td>accept request (g)</td>
<td>19-3-2014 6:48</td>
<td>Mary</td>
</tr>
</tbody>
</table>
Play Out: A possible scenario

<table>
<thead>
<tr>
<th>Case</th>
<th>Activity</th>
<th>Timestamp</th>
<th>Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>432</td>
<td>register travel request (a)</td>
<td>18-3-2014:9.15</td>
<td>John</td>
</tr>
<tr>
<td>432</td>
<td>get support from local manager (b)</td>
<td>18-3-2014:9.25</td>
<td>Mary</td>
</tr>
<tr>
<td>432</td>
<td>check budget by finance (d)</td>
<td>19-3-2014:8.55</td>
<td>John</td>
</tr>
<tr>
<td>432</td>
<td>decide (e)</td>
<td>19-3-2014:9.36</td>
<td>Sue</td>
</tr>
<tr>
<td>432</td>
<td>accept request (g)</td>
<td>19-3-2014:9.48</td>
<td>Mary</td>
</tr>
</tbody>
</table>
Play Out: Another scenario

- register travel request (a)
- get support from local manager (b)
- get detailed motivation letter (c)
- check budget by finance (d)
- decide (e)
- accept request (g)
- reject request (h)
- reinitiate request (f)

Start end

Sequence: a d c e f b d e h
Play Out: Process model allows for many more scenarios

- **Start**
  - Register travel request (a)
  - Get support from local manager (b)
  - Get detailed motivation letter (c)
  - Check budget by finance (d)
  - Decide (e)
  - Accept request (g)
  - Reject request (h)
  - Reinitiate request (f)

**End**
Play-In
Play In: Simple process allowing for 4 traces

abdeg adbeg adbeg adbeg
abdeh abdeh abdeh abdeh
abdeh abdeh abdeh abdeh
abdeh abdeh abdeh abdeh
Play In:
Process allowing for more traces

[Diagram showing the process flow with nodes for register travel request, get support, get detailed motivation letter, check budget, decide, accept or reject request, and reinitiate request.]

- register travel request (a)
- get support from local manager (b)
- get detailed motivation letter (c)
- check budget by finance (d)
- decide (e)
- accept request (g)
- reject request (h)
- reinitiate request (f)
No modeling needed!
Example Process Discovery
(Dutch housing agency, 208 cases, 5987 events)
Example process discovery for hospital
(627 gynecological oncology patients, 24331 events)
Process discovery algorithms (small selection)

- automata-based learning
- heuristic mining
- genetic mining
- stochastic task graphs
- ETM genetic algorithm
- fuzzy mining
- mining block structures
  - \(\alpha\) algorithm
  - \(\alpha#\) algorithm
  - \(\alpha++\) algorithm
- distributed genetic mining
- language-based regions
- state-based regions
- LTL mining
- Inductive Miner (infrequent)
- neural networks
- hidden Markov models
- conformal process graph
- partial-order based mining
- ILP mining
Region $R = (X,Y,c)$ corresponding to place $p_R$: $X = \{a_1,a_2,c_1\}$ = transitions producing a token for $p_R$, $Y = \{b_1,b_2,c_1\}$ = transitions consuming a token from $p_R$, and $c$ is the initial marking of $p_R$. 
Basic idea: enough tokens should be present when consuming

A place is feasible if it can be added without disabling any of the traces in the event log.

for any $\sigma \in L, k \in \{1, \ldots, |\sigma|\}$, $\sigma_1 = h^{-1}(\sigma)$, $a = \sigma(k)$, $\sigma_2 = h^k(\sigma) = \sigma_1 \oplus a$:

$$c + \sum_{t \in X} \partial_{\text{multiset}}(\sigma_1)(t) - \sum_{t \in Y} \partial_{\text{multiset}}(\sigma_2)(t) \geq 0.$$
Replay

<table>
<thead>
<tr>
<th>Case</th>
<th>Activity</th>
<th>Timestamp</th>
<th>Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>432</td>
<td>register travel request (a)</td>
<td>18-3-2014 9:15</td>
<td>John</td>
</tr>
<tr>
<td>432</td>
<td>get support from local manager (b)</td>
<td>18-3-2014 9:25</td>
<td>Mary</td>
</tr>
<tr>
<td>432</td>
<td>check budget by finance (d)</td>
<td>19-3-2014 8:55</td>
<td>John</td>
</tr>
<tr>
<td>432</td>
<td>decide (e)</td>
<td>19-3-2014 9:36</td>
<td>Sue</td>
</tr>
<tr>
<td>432</td>
<td>accept request (g)</td>
<td>19-3-2014 9:48</td>
<td>Mary</td>
</tr>
</tbody>
</table>
Replay

a  c  e  g

check budget (d) is missing!
Alignments: Relating reality and model

```
(a) register travel request
(c) get detailed motivation letter
(b) get support from local manager
(d) check budget by finance
(e) decide
(g) accept request
(h) reject request
(f) reinitiate request
```

check budget (d) did not happen but should have according to the model.
register travel request (a)
get detailed motivation letter (c)
get support from local manager (b)
check budget by finance (d)
decide (e)
accept request (g)
reject request (h)
reinitiate request (f)
start
end

.reject request (h) is impossible
Alignments: Relating reality and model

a | c | h | d | e | g

a | c » d | e | g

reject request (h) happened but could not happen according to the model
Any trace in reality can be related to a path in the model
Any trace in reality can be related to a path in the model.

Optimization problem using a cost function:

- start
- check is missing
- get detailed motivation letter (c)
- check budget by finance (d)
- decide (e)
- accept request (g)
- reject request (h)
- end

Check is missing:
- one check too many
- cannot both be done

Reinitiate request (f)
Replay with timestamps

1. Register travel request (a) 9:15
2. Get detailed motivation letter (c) 9:20
3. Get support from local manager (b) 9:35
4. Check budget by finance (d) 10:15
5. Decide (e) 11:30
6. Accept request (g) 11:30
7. Reject request (h)
8. Reinitiate request (f) 9:35

Start 

End
Replay with timestamps for many traces

- register travel request (a)
- get detailed motivation letter (c)
- get support from local manager (b)
- check budget by finance (d)
- decide (e)
- accept request (g)
- reject request (h)
- reinitiate request (f)

- frequencies of activities
- durations of activities
- waiting times and other delays between activities
- frequencies of paths
Alignments are essential!

- conformance checking to diagnose deviations
- squeezing reality into the model to do model-based analysis
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_CANCELLED”, “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 (Replay project)
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_CANCELLED” and “W_CANCELLED”, “O_CREATED” and “W_CREATED”, “O_SELECTED” and “W_SELECTED”, “O_SENT” and “W_SENT” occurred with the same frequency value (i.e. 60) before parallel branch

Many moves on log of “W_Afhandelen leads” 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”

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

Move on log of “O_CANCELLED” and “A_CANCELLED”

Moves on model towards end of traces
“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 occurred rarely (only 4 times).

The average waiting time for the input place of “W_Nabellen offertes+START” is very long (2.83 days) compared to the average waiting time of other places.
Software
600+ plug-ins available covering the whole process mining spectrum
Overview: Role of process models

- "world" (people, machines, components, organizations)
- Business processes
- Models: specifies, configures, implements, analyzes
- Play-Out
- Play-In
- Replay
- Software system: records events, e.g., messages, transactions, etc.
- Event logs
- Conformance
- Enhancement
- Discovery
- Supports/controls
- Models: analyzes

(process) model
Big data
"DATA IS THE NEW OIL."

From the beginning of recorded time until 2003, we created 5 exabytes (5 billion gigabytes) of data.

In 2011 the same amount was created every two days.

By 2013, it's expected that the time will shrink to 10 minutes.

There are 133 million blogs on the web.

As of August 2012, there were just over 4 million articles in the English Wikipedia.

80% of all humans own something of some sort. Out of 5 billion mobiles, 1 billion are smartphones. On Singapore, 64% of citizens are smartphone users.

60% of all humans 6.4 billion people are activeators. In 2010, 105,000 text messages were sent every second.

10% of all photos ever taken were taken in 2011.

247 billion emails are sent every day. (Up to 80% are spam.)

50% of 5-year-old kids in the U.S. are given access to a smartphone.

With new fiber-optic cable, the round-trip time between New York and London will be 59.8 milliseconds.

This 5-millisecond saving is worth many millions of dollars to the banks who use the cable (and pay millions to do so).

How they save 5 million.

The depth of the Atlantic.

The new cable will run along the floor that is up to 1 mile below the current floor, a different route. It is shorter, making the journey faster along the way.

The new cable is expected to cut the cost of sending data by 50%.
What if?

there are more than 100,000,000 events?

there are more than 1000 different activities?

there are more than 1,000,000 cases?
Decompose event log!
vertical or horizontal

sets of cases
sets of activities
Vertical distribution: Split cases

sets of cases

sets of cases

sets of cases

sets of cases

sets of cases
Horizontal distribution

sets of activities

{a, b, e, f, g}

{b, c, d, e}

- abeg
- abefbeg
- abeg
- abefbeg
- abeg
- abefbeg
- abeg
- abefbeg
- abeg
- abefbeg
- abeg
- abefbeg
- abeg
- abefbeg
- abeg
- abefbeg
- abeg

- bcde
- bdcebcde
- bdce
- bcdebcde
- bdcebdce
- bcdebdce
- bcde
- bdce
- bcdebdce
- bcdebdce
- bcde
- bdce
- bcde
Horizontal distribution: The key idea

projected on \{a,b,e,f,g\}

projected on \{b,c,d,e\}
Two foundational ways of spitting event data: horizontal or vertical.
Decomposing Conformance Checking

- Decomposition technique
- Conformance checking technique

- **SN** process model
  - Decompose model
  - L event log

- M^1 submodel
  - Conformance check
  - L_1 sublog

- M^2 submodel
  - Conformance check
  - L_2 sublog

- M^n submodel
  - Conformance check
  - L_n sublog

E.g., maximal decomposition, passage-based decomposition, or SESE/RPST-based decomposition.

Yields a (valid) activity partitioning.

E.g., A* based alignments, token-based replay, or simple replay until first deviation.

See "divide and conquer" framework by Eric Verbeek.
Example of a valid decomposition

Log can be split in the same way!
Example of an alignment for observed trace a,b,c,d,e,c,d,g,f

\[ \gamma_3 = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\ a & b & c & d & e & c & d & e & c & \tau & d & \tau & g & f & t1 \\ a & b & c & d & e & c & \tau & d & \tau & g & f & t1 \end{bmatrix} \]

Etc.
Conformance checking can be decomposed !!!

- General result for any valid decomposition: Any event log or trace is perfectly fitting the overall model if and only if it is also fitting all the individual fragments.

Example
(work with Jorge Munoz-Gama and Josep Carmona)

**Diagnose Subprocess**
Detect an unfitting subprocess, analyze it in isolation, and diagnose the cause of the problems.

**Diagnose Non Fitting Net**
Detect all unfitting subprocesses, compose the net that contains all them, and diagnose the cause of the problems.
Decomposing Process Discovery

- Event log \( L \) is decomposed using causal graph based on frequencies, SESE/RPST, or language/state-based region discovery, variants of alpha algorithm, genetic process mining, yielding a (valid) activity partitioning.

- Process model \( M \) is divided into submodels \( M_1, M_2, \ldots, M_n \), each corresponding to a sublog \( L_1, L_2, \ldots, L_n \). Each submodel undergoes process discovery.

See "divide and conquer" framework by Eric Verbeek.
conclusion
Process mining: mediating between modeled and observed behavior

Decomposition as a way to deal with "Big" process mining tasks

Many challenges, e.g., process discovery
More and more information about business processes is recorded by information systems in the form of so-called "event logs". Despite the omnipresence of such data, most organizations diagnose problems based on fiction rather than facts. Process mining is an emerging discipline based on process model-driven approaches and data mining. It not only allows organizations to fully benefit from the information stored in their systems, but it can also be used to check the conformance of processes, detect bottlenecks, and predict execution problems.

Wil van der Aalst delivers the first book on process mining. It aims to be self-contained while covering the entire process mining spectrum from process discovery to operational support. In Part I, the author provides the basics of business process modeling and data mining necessary to understand the remainder of the book. Part II focuses on process discovery as the most important process mining task. Part III moves beyond discovering the control flow of processes and highlights conformance checking, and organizational and time perspectives. Part IV guides the reader in successfully applying process mining in practice, including an introduction to the widely used open-source tool ProM. Finally, Part V takes a step back, reflecting on the material presented and the key open challenges.

Overall, this book provides a comprehensive overview of the state of the art in process mining. It is intended for business process analysts, business consultants, process managers, graduate students, and BPM researchers.

Features and Benefits:
- First book on process mining, bridging the gap between business process modeling and business intelligence.
- Written by one of the most influential and most-cited computer scientists and the best-known BPM researcher.
- Self-contained and comprehensive overview for a broad audience in academia and industry.
- The reader can put process mining into practice immediately due to the applicability of the techniques and the availability of the open-source process mining software ProM.
Process Mining: Data Science in Action

https://www.coursera.org/course/procmin

First Massive Open Online Course (MOOC) on Process Mining

About the Course

Data science is the profession of the future, because organizations that are unable to use (big) data in a smart way will not survive. It is not sufficient to focus on data storage and data analysis. The data scientist also needs to relate data to process analysis. Process mining bridges the gap between traditional model-based process analysis (e.g., simulation and other business process management techniques) and data-centric analysis techniques such as machine learning and data mining. Process mining seeks the connection between event data (i.e., observed behavior) and process models (hand-made or discovered automatically).

This technology has become available only recently, but it can be applied to any type of operational processes (organizations and systems). Example applications include: analyzing treatment processes in hospitals, improving customer service processes in a multinational, understanding the browsing behavior of customers using a booking site, analyzing failures of a baggage handling system, and improving the user interface of an X-ray machine. All of these applications have in common that dynamic behavior needs to be related to process models. Hence, we refer to this as "data science in action".

The course explains the key analysis techniques in process mining. Participants will learn various process discovery algorithms. These can be used to automatically learn process models from raw event data. Various other process analysis techniques that use event data will be presented. Moreover, the course will provide easy-to-use software, real-life data sets, and practical skills to directly apply the theory in industrial engineering.

Sessions

- Starts in 3 months
- Nov 12th 2014 - Dec 24th 2014

Eligible for

Statement of Accomplishment

Course at a Glance

- 5 weeks of study
- 4-6 hours of work / week
- English
- English subtitles

Instructors

- Wil van der Aalst
  Eindhoven University of Technology

Categories

- Data mining
- Business process intelligence
- Large scale distributed computing
- Visual analytics
- Industrial engineering
- Behavioral/social sciences
- Data science
- Machine learning
- Statistics
- Stochastics
- Algorithms
- Databases
- Domain knowledge
- Table visualizations
- Process mining