FOCUS ON PROCESS MINING ALGORITHMS

Paolo Ceravolo
paolo.ceravolo@unimi.it

Dipartimento di Informatica - Sede di Crema
Thursday 14.00 - 15.00
Outline of the course

1. Definition of BPI
2. Overview on BI methods
3. Focus on Process Mining algorithms
4. Requisites for a well founded BI project
5. Exercise: review BPI works
Industry has developed several standards for representing business process data:

- **UML activity diagram**
- **Business Process Model Notation**
- **Event-driven Process Chain**
  - They *do not have an execution semantics*
  - It is often assumed they require a non-local semantics
Petri Nets

Petri Nets have an exact mathematical definition of their execution semantics

A **Petri Net** is a **directed bipartite graph**

- **Transitions** - event that occurs - bars
- **States or Places** - conditions to events

Arcs run from an input place to a transition or from a transition and an output place, never between places or between transitions
Petri Nets

Petri nets have an exact mathematical definition of their execution semantics.

The directed arcs describe which places are pre- and/or post- conditions for which transitions.

Places may contain a discrete number of marks called tokens. Any distribution of tokens over the places will represent a configuration of the net called a marking.
Petri Nets

US Ship, the Maersk Alabama, 350 miles off Somalia’s coast

Pirates in waters off of Somalia's coast

Pirates attack US Ship

Pirates have US Captain hostage

At tempted trade of hostages

Captain hostage aboard pirate’s boat

Capt a in hostage aboard pirate’s boat

Navy snipers rescue Captain

3 pirates dead, 1 taken into custody

Capt ain is free

Rest of Maersk crew is free

Maersk crew has one pirate hostage
FIGURE 24.3 Petri net primitives to represent system features. (a) Sequential, (b) conflict, (c) concurrent, (d) synchronization, (e) mutual exclusive, and (f) priority.
Model-less Data

In the past the focus has been mostly on process modelling and automation.

But often there is no model, or only a very primitive process model is available.

However the Information Systems record logs including a lot of implicit information on process execution (date, time, activity, staff).
Process Mining

Combining appropriate Data Mining techniques with Event Logs we can get insight about the executed processes

**Observed Process vs Expected Process**

- Discovery Process Model
- Check Conformance of Event Logs over a Model
- Discovery dysfunctional behaviour
  - Bottlenecks, waiting time, non alternatives
  - Help identifying their causes
- Optimise future executions
Process Mining
PM: Scenarios

Trends in information systems:
1. From programming to assembling
2. From data orientation to process orientation
3. From design to redesign and organic growth

PM: Scenarios

Process mining is designed to support Business Process Management, i.e. understand or monitor the control flow of a Business Process: the order in which individual activities are executed.

Typical patterns may be:
synchronisation, parallelisation, loops or combinations
PM: Scenarios

• When the process execution is related to quality metrics we can study how process behaviour impact on performances
  – Defining Metrics
  – Extracting Data
  – Characterising Process Behaviour
Connecting Business Process Model to observed Business Process Executions may also be relevant for measuring Key Performance Indicators, such as for instance:

- Average process overdue time
- Percentage of overdue processes
- Average time to complete task
- Percentage of processes where the actual number assigned resources is less than planned number of assigned resources
- Sum of costs of “killed” / stopped active processes
- Volume of tasks per staff
- Number of customer complaints
- Number of process errors vs Number of human errors
- Time allocated for administration, management, training
PM: Models

A Petri net which models a business process definition is called a WorkFlow net (WF-net).

A WF-net has one input place (start) and one output place (end).

In addition a WF-net has to satisfy three requirements:

1. *Option to complete*: it is always possible to reach a transition that marks place end.
2. *Proper Completion*: if place end is marked all other places are empty.
3. *No dead transition*: it should be possible to execute an arbitrary transition by following the appropriate route through the WF-net.
Process Mining algorithms interpret an Event Log as a multiset of traces and infer models by unifying these traces.

The IEEE Task Force on Process Mining has proposed a standard to describe event logs and event streams: the eXtensible Event Stream (XES)

An event is a quadruple $e = (c, a, r, t) \in E$, denoting the occurrence of an activity $a$ in a case $c$, using the resource $r$ at time $t$. The event universe can be indicated as the combination of all these elements: $E = C \times A \times R \times T$.

A trace is $\sigma \in A^*$ is a finite sequence of activities

A case is $\sigma \in E^*$ is a finite sequence of events related to the same case. This notion is useful as different cases can refer to the same trace.
**PM: Event Log**

<table>
<thead>
<tr>
<th>Case ID</th>
<th>Task Name</th>
<th>Event Type</th>
<th>Originator</th>
<th>Timestamp</th>
<th>Extra Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>File Fine</td>
<td>Completed</td>
<td>Anne</td>
<td>20-07-2004 14:00:00</td>
<td>...</td>
</tr>
<tr>
<td>2</td>
<td>File Fine</td>
<td>Completed</td>
<td>Anne</td>
<td>20-07-2004 15:00:00</td>
<td>...</td>
</tr>
<tr>
<td>1</td>
<td>Send Bill</td>
<td>Completed</td>
<td>system</td>
<td>20-07-2004 15:05:00</td>
<td>...</td>
</tr>
<tr>
<td>2</td>
<td>Send Bill</td>
<td>Completed</td>
<td>system</td>
<td>20-07-2004 15:07:00</td>
<td>...</td>
</tr>
<tr>
<td>3</td>
<td>File Fine</td>
<td>Completed</td>
<td>Anne</td>
<td>21-07-2004 10:00:00</td>
<td>...</td>
</tr>
<tr>
<td>3</td>
<td>Send Bill</td>
<td>Completed</td>
<td>system</td>
<td>21-07-2004 14:00:00</td>
<td>...</td>
</tr>
<tr>
<td>4</td>
<td>File Fine</td>
<td>Completed</td>
<td>Anne</td>
<td>22-07-2004 11:00:00</td>
<td>...</td>
</tr>
<tr>
<td>4</td>
<td>Send Bill</td>
<td>Completed</td>
<td>system</td>
<td>22-07-2004 11:10:00</td>
<td>...</td>
</tr>
<tr>
<td>1</td>
<td>Process Payment</td>
<td>Completed</td>
<td>system</td>
<td>24-07-2004 15:05:00</td>
<td>...</td>
</tr>
<tr>
<td>1</td>
<td>Close Case</td>
<td>Completed</td>
<td>system</td>
<td>24-07-2004 15:06:00</td>
<td>...</td>
</tr>
<tr>
<td>2</td>
<td>Send Reminder</td>
<td>Completed</td>
<td>Mary</td>
<td>20-08-2004 10:00:00</td>
<td>...</td>
</tr>
<tr>
<td>3</td>
<td>Send Reminder</td>
<td>Completed</td>
<td>John</td>
<td>21-08-2004 10:00:00</td>
<td>...</td>
</tr>
<tr>
<td>2</td>
<td>Process Payment</td>
<td>Completed</td>
<td>system</td>
<td>22-08-2004 09:05:00</td>
<td>...</td>
</tr>
<tr>
<td>2</td>
<td>Close case</td>
<td>Completed</td>
<td>system</td>
<td>22-08-2004 09:06:00</td>
<td>...</td>
</tr>
<tr>
<td>4</td>
<td>Send Reminder</td>
<td>Completed</td>
<td>John</td>
<td>22-08-2004 15:10:00</td>
<td>...</td>
</tr>
<tr>
<td>4</td>
<td>Send Reminder</td>
<td>Completed</td>
<td>Mary</td>
<td>22-08-2004 17:10:00</td>
<td>...</td>
</tr>
<tr>
<td>4</td>
<td>Process Payment</td>
<td>Completed</td>
<td>system</td>
<td>29-08-2004 14:01:00</td>
<td>...</td>
</tr>
<tr>
<td>4</td>
<td>Close Case</td>
<td>Completed</td>
<td>system</td>
<td>29-08-2004 17:30:00</td>
<td>...</td>
</tr>
<tr>
<td>3</td>
<td>Send Reminder</td>
<td>Completed</td>
<td>John</td>
<td>21-09-2004 10:00:00</td>
<td>...</td>
</tr>
<tr>
<td>3</td>
<td>Send Reminder</td>
<td>Completed</td>
<td>John</td>
<td>21-10-2004 10:00:00</td>
<td>...</td>
</tr>
<tr>
<td>3</td>
<td>Process Payment</td>
<td>Completed</td>
<td>system</td>
<td>25-10-2004 14:00:00</td>
<td>...</td>
</tr>
<tr>
<td>3</td>
<td>Close Case</td>
<td>Completed</td>
<td>system</td>
<td>25-10-2004 14:01:00</td>
<td>...</td>
</tr>
</tbody>
</table>
The first algorithm proposed in PM is the $\alpha$-algorithm

- $L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\}$
- $a \succ_L b$ if and only if there is a trace $\sigma = t_1, t_2, t_3, \ldots, t_n$ and $i \in \{1, \ldots, n-1\}$ such that $\sigma \in L$ and $t_i = a$ and $t_{i+1} = b$
- $a \rightarrow_L b$ if and only if $a \succ_L b$ and $b \nLeftarrow_L a$
- $a \Leftarrow_L b$ if and only if $a \nLeftarrow_L b$ and $b \nLeftarrow_L a$
- $a \|_L b$ if and only if $a \succ_L b$ and $b \succ_L a$
PM: Discovery Techniques

• $L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\}$
• $a \succ b, a \succ c, a \succ e \quad \rightarrow \quad b \nless a, c \nless a, e \nless a$
PM: Discovery Techniques

• $L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\}$
• $a \succ b, a \succ c, a \succ e \implies b \succ a, c \succ a, e \succ c$
• $a \rightarrow b, a \rightarrow c, a \rightarrow e$
PM: Discovery Techniques

- $L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\}$
- $a \succ b, a \succ c, a \succ e \quad \rightarrow \quad a \Uparrow b, a \Uparrow c, a \Uparrow e$
- $a \rightarrow b, a \rightarrow c, a \rightarrow e$
- $b \succ c, b \succ d, b \Uparrow e \quad \rightarrow \quad c \succ b, d \Uparrow b \quad \rightarrow \quad e \Uparrow b$
PM: Discovery Techniques

• $L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\}$
• $a \succ b, a \succ c, a \succ e \quad \rightarrow \quad a \nless b, a \nless c, a \nless e$
• $a \rightarrow b, a \rightarrow c, a \rightarrow e$
• $b \succ c, b \succ d, b \nless e \quad \rightarrow \quad c \succ b, d \nless b \quad \rightarrow \quad e \nless b$
• $b \parallel c, b \rightarrow d, b \nnot= e$
PM: Discovery Techniques

- \( L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\} \)
- \( a \succ b, a \succ c, a \succ e \quad \rightarrow \quad a \nrightarrow b, a \nrightarrow c, a \nrightarrow e \)
- \( a \rightarrow b, a \rightarrow c, a \rightarrow e \)
- \( b \succ c, b \succ d, b \nrightarrow e \quad \rightarrow \quad c \succ b, d \nrightarrow b \quad \rightarrow \quad e \nrightarrow b \)
- \( b \parallel c, b \rightarrow d, b \nright# e \)
- \( c \succ d, c \succ b, c \nrightarrow e \quad \rightarrow \quad b \succ c, d \nrightarrow c \quad \rightarrow \quad e \nrightarrow c \)
PM: Discovery Techniques

• \( L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\} \)
• \( a \succ b, a \succ c, a \succ e \quad \rightarrow \quad a \not\succ b, a \not\succ c, a \not\succ e \)
• \( a \rightarrow b, a \rightarrow c, a \rightarrow e \)
• \( b \succ c, b \succ d, b \not\succ e \quad \rightarrow \quad c \succ b, d \not\succ b \quad \rightarrow \quad e \not\succ b \)
• \( b \parallel c, b \rightarrow d, b \# e \)
• \( c \succ d, c \succ b, c \not\succ e \quad \rightarrow \quad b \succ c, d \not\succ c \quad \rightarrow \quad e \not\succ c \)
• \( c \parallel b, c \rightarrow d, c \# e \)
PM: Discovery Techniques

• $L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\}$
• $a \succ b, a \succ c, a \succ e \rightarrow a \not\succ b, a \not\succ c, a \not\succ e$
• $a \rightarrow b, a \rightarrow c, a \rightarrow e$
• $b \succ c, b \succ d, b \not\succ e \rightarrow c \succ b, d \not\succ b \rightarrow e \not\succ b$
• $b \parallel c, b \rightarrow d, b \not\# e$
• $c \succ d, c \succ b, c \not\succ e \rightarrow b \succ c, d \not\succ c \rightarrow e \not\succ c$
• $c \parallel b, c \rightarrow d, c \not\# e$
• $e \succ d \rightarrow d \not\succ e$
PM: Discovery Techniques

• $L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\}$
• $a \succ b, a \succ c, a \succ e$ — $a \not\succ b, a \not\succ c, a \not\succ e$
• $a \rightarrow b, a \rightarrow c, a \rightarrow e$
• $b \succ c, b \succ d, b \not\succ e$ — $c \succ b, d \not\succ b$ — $e \not\succ b$
• $b \parallel c, b \rightarrow d, b \# e$
• $c \succ d, c \succ b, c \not\succ e$ — $b \succ c, d \not\succ c$ — $e \not\succ c$
• $c \parallel b, c \rightarrow d, c \# e$
• $e \succ d$ — $d \not\succ e$
• $e \rightarrow d$
PM: Discovery Techniques

• $L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\}$

• $a \rightarrow b, a \rightarrow c, a \rightarrow e$

• $b || c, b \rightarrow d, b \# e$

• $c || b, c \rightarrow d, c \# e$

• $e \rightarrow d$
PM: Discovery Techniques

• $L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\}$
• $a \rightarrow b, a \rightarrow c, a \rightarrow e$
PM: Discovery Techniques

- $L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\}$
- $a \rightarrow b, a \rightarrow c, a \rightarrow e$
- $b \parallel c, b \# e$
- $c \parallel b, c \# e$
PM: Discovery Techniques

- $L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\}$
- $a \rightarrow b, a \rightarrow c, a \rightarrow e$
- $b \parallel c, b \rightarrow d, b \# e$
- $c \parallel b, c \rightarrow d, c \# e$
- $e \rightarrow d$
PM: Discovery Techniques

- $L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\}$
- $a \rightarrow b, a \rightarrow c, a \rightarrow e$
- $b \parallel c, b \rightarrow d, b \# e$
- $c \parallel b, c \rightarrow d, c \# e$
- $e \rightarrow d$
PM: Discovery Techniques

- $L_2 = \{a, b, c, d\}, \{a, c, b, d\}, \{a, e, d\}$
- $a \rightarrow b, a \rightarrow c, a \rightarrow e$
- $b||c, b \rightarrow d, b\#e$
- $c||b, c \rightarrow d, c\#e$
- $e \rightarrow d$
PM: Discovery Techniques

- The **heuristic miner** introduced the filtering of infrequent behaviour:
  
  $$a \rightarrow_L b = |a >_L b| - |b >_L a| / |a >_L b| + |b >_L a| + 1$$

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>0</td>
<td>11</td>
<td>11</td>
<td>13</td>
<td>5</td>
</tr>
<tr>
<td>b</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>c</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>d</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>13</td>
</tr>
<tr>
<td>e</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
PM: Discovery Techniques

(a) sequence pattern: a→b

(b) XOR-split pattern:
   a→b, a→c, and b#c

(c) XOR-join pattern:
   a→c, b→c, and a#b

(d) AND-split pattern:
   a→b, a→c, and b||c

(e) AND-join pattern:
   a→c, b→c, and a||b
PM: Discovery Techniques

• The **genetic miner** applies the basic principles of the evolutionary algorithms:
  • Given an initial solution (a model) and a goal evolve this solution for optimising this goal
  • Genotype: causal matrices containing the input and output activities of each transition
  • Goal: fitness function
  • Evolution: mutation and crossover
PM: Discovery Techniques

$L_3 = \{a, c, d, e, h\}, \{a, b, d, e, g\}, \{a, d, c, e, h\}, \{a, d, b, e, h\}, \{a, c, d, e, g\}, \{a, d, c, e, g\}, \{a, f, c, e, g\}, \ldots$
PM: Discovery Techniques

(all 21 variants seen in the log)
PM: Conformance Checking

Conformance Checking is a supervised learning technique that requires as input both an event log and a model.

• Evaluate the quality of a model
• Evaluate the deviations of a log from the model
  • It can be used for auditing and business process compliance
PM: Conformance Checking

To compute the degree of conformance of a Model over an Event Log the traces in E are replayed over M, the following variables are typically measured:

- \( n \) the number of activities in the process
- \( m \) the number of missing tokens
- \( c \) the number of consumed tokens
- \( r \) the number of remaining tokens
- \( p \) the number of produced tokens

\[
f = \frac{1}{2} \left( 1 - \frac{\sum_{i=1}^{k} n_i m_i}{\sum_{i=1}^{k} n_i c_i} \right) + \frac{1}{2} \left( 1 - \frac{\sum_{i=1}^{k} n_i r_i}{\sum_{i=1}^{k} n_i p_i} \right)
\]
Let’s for instance replay the trace $t_i: \{a, c, e\}$
Rozinat, Anne and van der Aalst, Wil MP, Conformance checking of processes based on monitoring real behavior, 2008
Rozinat, Anne and van der Aalst, Wil MP, Conformance checking of processes based on monitoring real behaviyo, 2008
Fitness = 1− cost for aligning model and event log /
Minimal cost to align arbitrary event log on model and vice versa

Precision = # visited markings / #total marking visits over all markings

Generalization = average #nodes in event log / #nodes in the model

Simplicity= 1 – #duplicate activities + #missing activities / #event in event log

Rozinat, Anne and van der Aalst, Wil MP, Conformance checking of processes based on monitoring real behavior, 2008
.frame

PM: Evaluation

Comparing a PM approach with an automatic approach generation BPM from source code lead to controversial results

Pérez-Castillo et al An Empirical Comparison of Static and Dynamic Business Process Mining 2011
PM: Evaluation

Comparing a PM approach with an automatic approach generation BPM from source code lead to controversial results.

Pérez-Castillo et al. An Empirical Comparison of Static and Dynamic Business Process Mining 2011
PM: Extension Techniques

• When the process execution is related to quality metrics we can study how process behaviour impact on performances
  – Defining Metrics
  – Extracting Data
  – Characterising Process Behaviour
PM: Extension Techniques

- Waiting time
PM: Extension Techniques

- Decision Point Analysis
PM: Extension Techniques

• Business Impact Analysis
PM: Tools

- **ProM Family** [http://www.processmining.org/tools/start](http://www.processmining.org/tools/start)
- **My Invenio** [https://www.my-invenio.com/#body](https://www.my-invenio.com/#body)
PM: Extension Techniques

• It is easy if
  – You know the metrics to be tested
  – These metrics apply on data of the Event Log
  – Your models does not bias your learning procedures
  – Your user understand these metrics

• It is not just a feature selection problem
• It is not just a combination but a permutation
Interpretation
Interpretation

• We are dealing with a complex research spaces

• We are dealing with:
  • metrics that are compensative by nature
  • bias generated by complex algorithms
  • user may not understand their implications
KITE.IT project

- Knowledge and Business Intelligence Technologies in cross-Enterprise environments for Italian Advanced Mechanical Industry
  - Funded by the Italian Ministry of Economic Development under the “Industria 2015” contract
  - To introduce Process Intelligence into the aerospace industry
  - Prolong descriptive analysis with prescriptive and predictive scope

- Final goal: derive previously unknown and potentially unexpected knowledge
KITE approach

Methodology for Knowledge Acquisition:
• Metrics first
  – First restrict your data then apply PI (process cubes)
  – Exploit user knowledge: Business Rules
  – Use of process data + any auxiliary data (graph database)
• Descriptive knowledge: select your view on data
• Prescriptive knowledge: evaluate achievement of B.R.
  – Partition the KB according to Business Rules violations
• Predictive knowledge: incidence of specific properties on the subset of violations
KITE.IT Data Model: Zeus

- Extremely **generic** data representation model and easily extended with any **domain specific** info (aerospace)
- Atomic elements of RDF (Resource Description Framework): Graph as a set of **triples** (statements, tuples)
  - 3 elements (resource, relations between resource, attributes of resources)
  - Elements modelled as “labelled oriented graphs”: 
    `<s,p,o>` subject predicate object
  - Additive
  - Supporting continuous query
KITE.IT Data Model: Zeus
### Case Study: maintenance process

**Workflow of the engine maintenance process**

<table>
<thead>
<tr>
<th></th>
<th>Inspect</th>
<th>Disassembly</th>
<th>Inspect Module</th>
<th>Repair Module</th>
<th>Clean</th>
<th>Assembly</th>
<th>Bench Test</th>
<th>Check Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Revision</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td></td>
<td>YES</td>
</tr>
<tr>
<td>Minor Revision</td>
<td>YES</td>
<td></td>
<td>YES</td>
<td>YES</td>
<td></td>
<td></td>
<td>YES</td>
<td>YES</td>
</tr>
<tr>
<td>Inspect Only</td>
<td>YES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>YES</td>
</tr>
<tr>
<td>Out of Order</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
</tr>
</tbody>
</table>
New constraint i.e. new Business Rule:
In case of military airline engine all the maintenance operations must be performed by internal staff (no outsourcing)

Add new auxiliary data
Case Study: maintenance process

Check which rows are violating this new Business Rule

In case of military airline engine all the maintenance operations must be performed by internal staff (no outsourcing)
Case Study: maintenance process

Original Partition violating BR #1

<table>
<thead>
<tr>
<th>Process Id</th>
<th>Operations_type</th>
<th>Total Duration</th>
<th>Violating BR#1</th>
<th>Customer Type</th>
<th>Staff (Internal/Outsourced)</th>
<th>Violating BR#2</th>
<th>Initials</th>
<th>Company (if external)</th>
<th>Certified staff?</th>
</tr>
</thead>
<tbody>
<tr>
<td>p101</td>
<td>General Revision</td>
<td>12</td>
<td>Yes</td>
<td>Military</td>
<td>Internal</td>
<td>No</td>
<td>C.A.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p103</td>
<td>General Revision</td>
<td>13</td>
<td>Yes</td>
<td>Military</td>
<td>Internal</td>
<td>No</td>
<td>A.B.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p105</td>
<td>General Revision</td>
<td>12</td>
<td>Yes</td>
<td>Military</td>
<td>Internal</td>
<td>No</td>
<td>A.A.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p106</td>
<td>General Revision</td>
<td>13</td>
<td>Yes</td>
<td>Military</td>
<td>Internal</td>
<td>No</td>
<td>A.B.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p107</td>
<td>General Revision</td>
<td>12</td>
<td>Yes</td>
<td>Military</td>
<td>Internal</td>
<td>No</td>
<td>F.Z.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p11</td>
<td>Minor Revision</td>
<td>8</td>
<td>Yes</td>
<td>Military</td>
<td>Outsourced</td>
<td>Yes</td>
<td>T.R.</td>
<td>AB services</td>
<td>No</td>
</tr>
<tr>
<td>p09</td>
<td>Minor Revision</td>
<td>5</td>
<td>No</td>
<td>Civil</td>
<td>Internal</td>
<td>No</td>
<td>F.Z.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p1011</td>
<td>Out of Order</td>
<td>11</td>
<td>No</td>
<td>Civil</td>
<td>Outsourced</td>
<td>No</td>
<td>GP.G.</td>
<td>SerCo</td>
<td>Yes</td>
</tr>
<tr>
<td>p1015</td>
<td>Out of Order</td>
<td>14</td>
<td>No</td>
<td>Civil</td>
<td>Internal</td>
<td>No</td>
<td>E.D.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p102</td>
<td>General Revision</td>
<td>11</td>
<td>No</td>
<td>Civil</td>
<td>Internal</td>
<td>No</td>
<td>E.D.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p1021</td>
<td>Out of Order</td>
<td>11</td>
<td>No</td>
<td>Civil</td>
<td>Internal</td>
<td>No</td>
<td>E.D.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p1029</td>
<td>Out of Order</td>
<td>10</td>
<td>No</td>
<td>Civil</td>
<td>Outsourced</td>
<td>No</td>
<td>GP.G.</td>
<td>SerCo</td>
<td>Yes</td>
</tr>
<tr>
<td>p104</td>
<td>General Revision</td>
<td>11</td>
<td>No</td>
<td>Civil</td>
<td>Internal</td>
<td>No</td>
<td>F.Z.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p108</td>
<td>General Revision</td>
<td>11</td>
<td>No</td>
<td>Civil</td>
<td>Internal</td>
<td>No</td>
<td>A.A.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p109</td>
<td>General Revision</td>
<td>10</td>
<td>No</td>
<td>Civil</td>
<td>Internal</td>
<td>No</td>
<td>F.Z.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p12</td>
<td>Minor Revision</td>
<td>3</td>
<td>No</td>
<td>Civil</td>
<td>Internal</td>
<td>No</td>
<td>A.B.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p200</td>
<td>General Revision</td>
<td>2</td>
<td>No</td>
<td>Civil</td>
<td>Outsourced</td>
<td>No</td>
<td>GP.G.</td>
<td>SerCo</td>
<td>Yes</td>
</tr>
<tr>
<td>p201</td>
<td>Inspect Only</td>
<td>2</td>
<td>No</td>
<td>Civil</td>
<td>Outsourced</td>
<td>No</td>
<td>GP.G.</td>
<td>SerCo</td>
<td>Yes</td>
</tr>
<tr>
<td>p202</td>
<td>Inspect Only</td>
<td>2</td>
<td>No</td>
<td>Military</td>
<td>Internal</td>
<td>No</td>
<td>A.B.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p2301</td>
<td>Out of Order</td>
<td>13</td>
<td>No</td>
<td>Civil</td>
<td>Internal</td>
<td>No</td>
<td>A.B.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p300</td>
<td>Out of Order</td>
<td>18</td>
<td>No</td>
<td>Civil</td>
<td>Internal</td>
<td>No</td>
<td>A.B.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p301</td>
<td>Out of Order</td>
<td>20</td>
<td>No</td>
<td>Military</td>
<td>Outsourced</td>
<td>Yes</td>
<td>T.R.</td>
<td>AB services</td>
<td>No</td>
</tr>
<tr>
<td>p31</td>
<td>Minor Revision</td>
<td>4</td>
<td>No</td>
<td>Civil</td>
<td>Internal</td>
<td>No</td>
<td>F.Z.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p33</td>
<td>Minor Revision</td>
<td>5</td>
<td>No</td>
<td>Military</td>
<td>Internal</td>
<td>No</td>
<td>A.B.</td>
<td>-</td>
<td>Yes</td>
</tr>
<tr>
<td>p39</td>
<td>Minor Revision</td>
<td>3</td>
<td>No</td>
<td>Civil</td>
<td>Internal</td>
<td>No</td>
<td>A.A.</td>
<td>-</td>
<td>Yes</td>
</tr>
</tbody>
</table>

SEE: incidence of new resources in “violating” partition
## Case Study: PLM system

<table>
<thead>
<tr>
<th>Partitions</th>
<th>Replace-based Fitness</th>
<th>Simple Behavior Appropriateness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entire Event Log</td>
<td>0.792</td>
<td>0.923</td>
</tr>
<tr>
<td>BR1: Approved by the head</td>
<td>0.9334</td>
<td>0.856</td>
</tr>
<tr>
<td>BR2: Viewed by &gt; 3 users</td>
<td>0.804</td>
<td>0.959</td>
</tr>
<tr>
<td>BR1 ∧ BR2</td>
<td>0.933</td>
<td>0.921</td>
</tr>
<tr>
<td>BR1 ∧ BR2 ∧ BR3</td>
<td>0.933</td>
<td>0.814</td>
</tr>
<tr>
<td>BR1 ∧ BR2 ∧ BR4</td>
<td>0.933</td>
<td>0.814</td>
</tr>
</tbody>
</table>
Conclusions

• Business Process Intelligence
  – First acquire knowledge on your process, then discover a model
  – This reduces your bias offer a better interpretation

• Complexity is an issue
  – Techniques allowing to limit complexity in local areas of the research space
  – BPI on data streams
The traditional PM approaches are proposed in the context of off-line data mining and cannot be used for dealing with large, growing amounts of data and for online analysis of process models.

Traditional approaches requires several iterations on an event log. To develop an online approach a runtime updated picture of the process behaviour is required.

This may be expressed in terms of constraints applying on the single event generated by the stream.
PM: Data Streams

Case: C1

Case: C2

Case: C3
# PM: Data Streams

<table>
<thead>
<tr>
<th>Existence templates</th>
<th>Template</th>
<th>Regular expression</th>
<th>Notation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cardinality templates</td>
<td>Participation(x)</td>
<td>[^x]<em>(([^x]</em>)^+[^x]*)</td>
<td><img src="image1" alt="Diagram" /></td>
</tr>
<tr>
<td></td>
<td>AtMostOne(x)</td>
<td>[^x]<em>(([^x]</em>))?[^x]*)</td>
<td><img src="image2" alt="Diagram" /></td>
</tr>
<tr>
<td>Position templates</td>
<td>Init(x)</td>
<td>[^x]<em>([^x]</em>)</td>
<td><img src="image3" alt="Diagram" /></td>
</tr>
<tr>
<td></td>
<td>End(x)</td>
<td>[^x]<em>([^x]</em>)</td>
<td><img src="image4" alt="Diagram" /></td>
</tr>
<tr>
<td>Forward-unidirectional relation templates</td>
<td>RespondedExistence((x, y))</td>
<td>[^x]<em>(([^x]</em>[^x]<em>[^x]</em>|^y[^x]<em>[^x]</em>)[^x]*)</td>
<td><img src="image5" alt="Diagram" /></td>
</tr>
<tr>
<td></td>
<td>Response((x, y))</td>
<td>[^x]<em>(([^x]</em>[^x]<em>[^x]</em>)[^x]*)</td>
<td><img src="image6" alt="Diagram" /></td>
</tr>
<tr>
<td></td>
<td>AlternateResponse((x, y))</td>
<td>[^x]<em>(([^x]</em>[^x]<em>[^x]</em>[^x]<em>)[^x]</em>)</td>
<td><img src="image7" alt="Diagram" /></td>
</tr>
<tr>
<td></td>
<td>ChainResponse((x, y))</td>
<td>[^x]<em>(([^x]</em>[^x]<em>[^x]</em>[^x]<em>)[^x]</em>)</td>
<td><img src="image8" alt="Diagram" /></td>
</tr>
</tbody>
</table>
### PM: Data Streams

<table>
<thead>
<tr>
<th>Relation templates</th>
<th>Precedence ((x, y))</th>
<th>AlternatePrecedence ((x, y))</th>
<th>ChainPrecedence ((x, y))</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>([-y])(x.<em>y)</em>[-y]*</td>
<td>([-y])(x[-y]<em>y[-y]<em>y[-y]</em>)((-y)) [-y]</em></td>
<td>([-y])(xy[-y]<em>y[-y]</em>)[-y]*</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Coupling templates</th>
<th>CoExistence ((x, y))</th>
<th>Succession ((x, y))</th>
<th>AlternateSuccession ((x, y))</th>
<th>ChainSuccession ((x, y))</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>([-xy])((x.<em>y)</em>((y.<em>x.</em>)) [-xy]*)</td>
<td>([-xy])(x.<em>y)[-xy]</em></td>
<td>([-xy])(x[-xy]<em>y[-xy]<em>y[-xy]</em>)[-xy]</em></td>
<td>([-xy])(xy[-xy]<em>y[-xy]</em>)[-xy]*</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Negative templates</th>
<th>NotChainSuccession ((x, y))</th>
<th>NotSuccession ((x, y))</th>
<th>NotCoExistence ((x, y))</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>([-x])(aa*[-xy]<em>[-x]</em>)((-x)*[-x])</td>
<td>([-x])(x[-y]<em>y[-xy]</em>)</td>
<td>([-xy])((x[-y]<em>y[-x]</em>)</td>
</tr>
</tbody>
</table>
PM: Data Streams

• The first problem to face is overfitting: if you simply generate the union of a set of observations with a large variability, rules will not get support.

• The support of a constraint is defined as the number of traces verifying the constraint divided by the total number of traces in the event log.

• The confidence value of a constraint scale the support by the percentage of traces in which the constraint is triggered, both parameters occur.

• IF you take support to 100% …
PM: Data Streams

<table>
<thead>
<tr>
<th>Backward-unidirectional relation templates</th>
<th>$\text{Precedence}(x, y)$</th>
<th>$\neg y^<em>(x</em>.y)[\neg y]^*$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\text{AlternatePrecedence}(x, y)$</td>
<td>$\neg y^<em>(x[\neg y]<em>y[\neg y]^</em>)[\neg y]^</em>$</td>
</tr>
<tr>
<td></td>
<td>$\text{ChainPrecedence}(x, y)$</td>
<td>$\neg y^<em>(xy[\neg y]^</em>)[\neg y]^*$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Coupling templates</th>
<th>$\text{CoExistence}(x, y)$</th>
<th>$\neg xy^<em>((x.<em>y.</em>)(y.<em>x.))</em>[\neg xy]^</em>$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\text{Succession}(x, y)$</td>
<td>$\neg xy^*(x.<em>y)[\neg xy]^</em>$</td>
</tr>
<tr>
<td></td>
<td>$\text{AlternateSuccession}(x, y)$</td>
<td>$\neg xy^<em>(x[\neg xy]<em>y[\neg xy]^</em>)[\neg xy]^</em>$</td>
</tr>
<tr>
<td></td>
<td>$\text{ChainSuccession}(x, y)$</td>
<td>$\neg xy^<em>(xy[\neg xy]^</em>)[\neg xy]^*$</td>
</tr>
</tbody>
</table>

| Negative templates                         | $\text{NotChainSuccession}(x, y)$ | $\neg x^*(aa*[\neg xy][\neg x]^*)*([\neg x]*|x)$ |
|-------------------------------------------|---------------------------|-----------------------------|
|                                           | $\text{NotSuccession}(x, y)$ | $\neg x^*(x[\neg y]^*)[\neg xy]^*$ |
|                                           | $\text{NotCoExistence}(x, y)$ | $\neg xy^*((x[\neg y]^*)|(y[\neg x]^*))?$ |

$t_1: \{x, c, y\}$
$t_1:\{x, c, y\}$, $t_2:\{x, y, c\}$
**PM: Data Streams**

<table>
<thead>
<tr>
<th>Backward-unidirectional relation templates</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Precedence</strong> ((x, y))</td>
</tr>
<tr>
<td><strong>AlternatePrecedence</strong> ((x, y))</td>
</tr>
<tr>
<td><strong>ChainPrecedence</strong> ((x, y))</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Coupling templates</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CoExistence</strong> ((x, y))</td>
</tr>
<tr>
<td><strong>Succession</strong> ((x, y))</td>
</tr>
<tr>
<td><strong>AlternateSuccession</strong> ((x, y))</td>
</tr>
<tr>
<td><strong>ChainSuccession</strong> ((x, y))</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Negative templates</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NotChainSuccession</strong> ((x, y))</td>
</tr>
<tr>
<td><strong>NotSuccession</strong> ((x, y))</td>
</tr>
<tr>
<td><strong>NotCoExistence</strong> ((x, y))</td>
</tr>
</tbody>
</table>

\(t_1:\{x, c, y\}, \ t_2:\{x, y, c\}, \ t_3:\{x, b, c\}\)
PM: Data Streams

• The second problem to face is **consistency**: using support less than 100% you may have conflicting constraints
  
  - `NotChainSuccession` (A PREACCEPTED, W Completeren aanvraag) and `ChainResponse` (A PREACCEPTED, W Completeren aanvraag)

• Such kind of incompatibilities must be eliminated
The third problem to face is **minimality**: using support less then 100% you may have constrains that in a relation of subsumption

- For example, `RespondedExistence(a,b)` states that if `a` occurs in a trace, then `b` has to occur in the same trace (either before or after `a`). `Response(a,b)` thus enforces `RespondedExistence(a,b)` by stating that not only must `b` be executed, but also that it must follow `a`. 
PM: Data Streams

Other Challenges

IEEE Transactions on Services Computing

Processes Meet Big Data: Connecting Data Science with Process Science

Wil van der Aalst ; Ernesto Damiani

Issue 6 • Date Nov.-Dec. 1 2015