

IWI – INSTITUT FÜR WIRTSCHAFTSINFORMATIK



Entwicklung einer Methode zum Process Mining unter besonderer Berücksichtigung von Organisationswissen

Master-Thesis

Tom Thaler

Inhaltsverzeichnis

| | |
|---|-------------|
| Abbildungsverzeichnis | V |
| Tabellenverzeichnis | VII |
| Abkürzungsverzeichnis | VIII |
| 1 Einleitung | 1 |
| 1.1 Motivation und Zielsetzung | 1 |
| 1.2 Gliederung der Arbeit | 2 |
| 2 Herangehensweise | 4 |
| 3 Grundlagen des Process Mining | 5 |
| 3.1 Begriffsklärung | 5 |
| 3.2 Sichten des Process Mining | 7 |
| 3.3 Einsatzmöglichkeiten | 9 |
| 3.4 Anforderungen an Verlaufsdaten | 11 |
| 3.5 Phasen im Process Mining | 13 |
| 3.5.1 Pre-Processing..... | 13 |
| 3.5.2 Processing | 13 |
| 3.5.3 Post-Processing | 14 |
| 3.6 Problemstellungen..... | 15 |
| 3.7 Mining-Methoden | 17 |
| 3.8 Ordnungsrahmen zum Process Mining | 18 |
| 3.9 Literaturübersicht und aktuelle Entwicklungen | 19 |
| 4 State-of-the-Art: Process Mining und Organisationswissen | 26 |
| 4.1 Einführung und Überblick..... | 26 |
| 4.2 Beziehungen zwischen Akteuren | 28 |
| 4.2.1 „Mining Social Networks“ | 28 |
| 4.2.2 Aktuelle Software..... | 32 |
| 4.3 Beziehungen zwischen Aufbau- und Ablauforganisation..... | 33 |
| 4.3.1 Grundlagen..... | 33 |
| 4.3.2 Potentiale des Organisationswissens..... | 34 |
| 4.3.3 Organisations-Metamodelle | 36 |
| 4.3.4 „Staff Assignment Mining“ | 41 |

| | | |
|----------|--|-----------|
| 4.3.5 | Aktuelle Software..... | 44 |
| 4.4 | Ableitung von Organisationsstrukturen aus Verlaufsdaten | 45 |
| 4.5 | Ableitung von Prozesswissen unter der Verwendung von Verlaufsdaten und Organisationswissen..... | 46 |
| 4.6 | Zusammenfassung und Fazit..... | 47 |
| 5 | Entwicklung einer Methode zum Process Mining unter der Verwendung von Organisationswissen | 49 |
| 5.1 | Einführung und Überblick..... | 49 |
| 5.1.1 | Zielsetzung | 49 |
| 5.1.2 | Anforderungen | 49 |
| 5.1.3 | Herangehensweise..... | 50 |
| 5.2 | Atomare Betrachtung von Aktivitäten als Problemstellung im Process Mining | 51 |
| 5.3 | Organisationswissen im Process Mining | 52 |
| 5.3.1 | Potentiale..... | 52 |
| 5.3.2 | Anforderungen an das Organisationsmodell..... | 59 |
| 5.3.3 | Organisations-Metamodell..... | 60 |
| 5.4 | Entwicklung der Mining-Methode..... | 61 |
| 5.4.1 | Anforderungen an die Verlaufsdaten | 61 |
| 5.4.2 | Auswahl des Zielmodells | 62 |
| 5.4.3 | Formalisierung der Ansätze | 63 |
| 5.4.4 | Umsetzung der vorgeschlagenen Potentiale..... | 67 |
| 6 | Prototyping | 69 |
| 6.1 | Einführung | 69 |
| 6.2 | ProM..... | 69 |
| 6.3 | Repräsentation der Verlaufsdaten | 69 |
| 6.4 | Repräsentation des Organisationsmodells..... | 71 |
| 6.5 | Auswahl der zu implementierenden Ansätze..... | 72 |
| 6.6 | Implementierung | 73 |
| 7 | Evaluation | 76 |
| 7.1 | Einführung und Vorgehensweise | 76 |
| 7.2 | Leginda..... | 76 |
| 7.2.1 | Vorstellung..... | 76 |

| | | |
|----------------------------------|---|--------------|
| 7.2.2 | Prozessbeschreibung | 77 |
| 7.3 | Vorbereitung der Evaluation | 78 |
| 7.3.1 | Anonymisierung | 78 |
| 7.3.2 | Separierung der Verlaufsdaten | 78 |
| 7.3.3 | Konvertierung der Verlaufsdaten nach MXML | 79 |
| 7.4 | Organizational Pre-Processing | 80 |
| 7.4.1 | Vorgehensweise | 80 |
| 7.4.2 | Version 2.20 | 80 |
| 7.4.3 | Version 2.30 | 81 |
| 7.4.4 | Version 2.40 | 82 |
| 7.4.5 | Ergebnisse | 82 |
| 7.5 | Processing | 84 |
| 7.5.1 | Vorgehensweise | 84 |
| 7.5.2 | Version 2.20 | 84 |
| 7.5.3 | Version 2.40 | 88 |
| 7.5.4 | Ergebnisse | 90 |
| 8 | Zusammenfassung und Fazit | 92 |
| 9 | Ausblick..... | 93 |
| Anhang | | X |
| | Literaturüberblick – Process Mining..... | X |
| | Literaturanalyse – Process Mining..... | XIV |
| Literaturverzeichnis..... | | XVIII |

Abbildungsverzeichnis

| | |
|---|----|
| Abb. 1: Informationsquelle des Process Mining | 6 |
| Abb. 2: Three types of process mining: (i) discovery, (ii) conformance, and (iii) extension ²³ | 10 |
| Abb. 3: Zeitlicher Verlauf der „Methoden“-Entwicklung | 21 |
| Abb. 4: Behandelte Sichten im Process Mining | 22 |
| Abb. 5: Zeitlicher Verlauf der Sichten-Bearbeitung | 22 |
| Abb. 6: Verwendete Mining-Methode | 23 |
| Abb. 7: Zeitlicher Verlauf der verwendeten Mining-Methoden | 23 |
| Abb. 8: Verwendete Modelltypen | 24 |
| Abb. 9: Zeitlicher Verlauf der verwendeten Modelltypen | 25 |
| Abb. 10: Vorgehen im Organizational Mining | 28 |
| Abb. 11: Soziogramm - "handover of work" | 31 |
| Abb. 12: EPK - Beispielprozess "Minig Social Networks" | 31 |
| Abb. 13: Soziogramm - "handover of work" mit Prozesswissen | 31 |
| Abb. 14: Screenshot – MiSoN | 33 |
| Abb. 15: Aufbau- und Ablauforganisation im ARIS-Haus | 34 |
| Abb. 16: Referenz-Metamodell der Aufbauorganisation | 41 |
| Abb. 17: Verwendetes Organisations-Metamodell im Staff Assignment Mining | 42 |
| Abb. 18: "Activity-Role-Performer"-Diagramm | 46 |
| Abb. 19: Tatsächliche und durch Log repräsentierte Parallelität | 53 |
| Abb. 20: Kundensegmentierung (EPK) | 58 |
| Abb. 21: Verwendetes Organisations-Metamodell | 60 |
| Abb. 22: XES 1.0 Standard Definition | 70 |
| Abb. 23: Prototyp: Import Organisationsmodell | 73 |
| Abb. 24: Prototyp: Organizational Pre-Processing Report | 75 |
| Abb. 25: Alpha Algorithmus - Leginda 2.20 – Original | 84 |
| Abb. 26: Alpha Algorithmus - Leginda 2.20 - OPP, Noise: 0 | 85 |
| Abb. 27: Alpha Algorithmus - Leginda 2.20 - OPP, Noise: 3 | 86 |
| Abb. 28: Genetic Miner - Leginda 2.20 – Original | 86 |
| Abb. 29: Genetic Miner - Leginda 2.20 - OPP, Noise: 0 | 87 |

| | |
|---|----|
| Abb. 30: Genetic Miner - Leginda 2.20 - OPP, Noise: 3 | 87 |
| Abb. 31: Alpha Algorithmus - Leginda 2.40 | 88 |
| Abb. 32: Genetic Miner - Leginda 2.40 – Original..... | 89 |
| Abb. 33: Genetic Miner - Leginda 2.40 - OPP, Noise: 3..... | 90 |

Tabellenverzeichnis

| | |
|---|------|
| Tabelle 1: Event-Log..... | 11 |
| Tabelle 2: Ordnungsrahmen zum Process Mining | 18 |
| Tabelle 3: Literaturkategorien und Anzahl bisheriger Arbeiten | 19 |
| Tabelle 4: Arbeiten in der Kategorie "Überblick" | 20 |
| Tabelle 5: "Mining Social Networks" - Beispiellog | 30 |
| Tabelle 6: Arbeiten zu Organisations-Metamodellen | 38 |
| Tabelle 7: Analyse - Organisations-Metamodelle..... | 39 |
| Tabelle 8: Beispiellog - Ableitung von Organisationsstrukturen aus Verlaufsdaten | 45 |
| Tabelle 9: Übersicht - Verlaufsdaten (Evaluation) | 79 |
| Tabelle 10: Aktionen des OPP - Version 2.20 - Parameter: 0 | 80 |
| Tabelle 11: Aktionen des OPP - Version 2.20 - Parameter: 3 | 81 |
| Tabelle 12: Aktionen des OPP - Version 2.30 - Parameter: 0 | 81 |
| Tabelle 13: Aktionen des OPP - Version 2.30 - Parameter: 3 | 81 |
| Tabelle 14: Literaturüberblick - Process Mining | XIV |
| Tabelle 15: Literaturanalyse - Process Mining | XVII |

Abkürzungsverzeichnis

| | |
|-------|--|
| ARIS | Architektur integrierter Informationssysteme |
| BAM | Business Activity Monitoring |
| BPA | Business Process Analysis |
| BPEL | Business Process Execution Language |
| BPM | Business Process Management |
| BPMS | Business Process Management System |
| BPMN | Business Process Modeling Notation |
| CFG | Control Flow Graph |
| CRM | Customer Relationship Management |
| CSV | Comma-separated values |
| DV | Datenverarbeitung |
| EIS | Enterprise Information Systems |
| eEPK | Erweiterte Ereignisgesteuerte Prozesskette |
| EPK | Ereignisgesteuerte Prozesskette |
| ERP | Enterprise Resource Planning |
| FSM | Finite State Machine |
| HTML | Hypertext Markup Language |
| IS | Informationssystem |
| IT | Information Technology |
| IuK | Informations- und Kommunikationssystem(e) |
| MiSoN | Mining Social Networks |
| OE | Organisationseinheit |
| OPP | Organizational Pre-Processing |
| PM | Process Mining |
| PPM | Process Performance Manager |
| QoS | Quality of Service |
| SCM | Supply Chain Management |
| SNA | Social Network Analysis |

| | |
|------|---------------------------------|
| SOA | Serviceorientierte Architektur |
| SOTA | State-of-the-Art |
| WfMC | Workflow Management Coalition |
| XES | Extensible Event Stream |
| XML | Extensible Markup Language |
| XSOM | XML Simple Organisational Model |

1 Einleitung

1.1 Motivation und Zielsetzung

Seit Anfang der 1990er Jahre hält die modellbasierte Ausführung von Prozessen immer mehr Einzug in Unternehmen und deren Informationssysteme. In der Software-Entwicklung werden heute beispielsweise vor der Implementierung Modelle der umzusetzenden Prozesse erstellt, sodass Missverständnissen vorgebeugt und die Erfolgsaussichten eines Projektes gesteigert werden können. Gleiches ist im betrieblichen Kontext zu erkennen – idealerweise werden Prozessmodelle entworfen, welche später in die Ausführung durch menschliche, maschinelle oder softwaretechnische Ressourcen überführt werden. BPM (= Business Process Management) oder Workflowmanagement-Systeme sind in der heutigen Zeit von BPMN 2.0 (= Business Process Modeling Notation) und BPEL (= Business Process Execution Language) sogar in der Lage, die grafisch visualisierten Modelle automatisch für die Ausführung aufzubereiten.

Weiterhin findet man heute in Unternehmen eine verteilte Systemarchitektur, in welcher IT-Komponenten gekapselt und anschließend wieder orchestriert werden (SOA = Serviceorientierte Architektur). Diese Architektur führt dazu, dass an einem Prozess viele unterschiedliche Informationssysteme, wie ERP- (Enterprise Resource Planing), CRM- (Customer Relationship Management) oder SCM-Systeme (Supply Chain Management) beteiligt sind, welche jeweils Informationen zu den einzelnen Prozessschritten bereitstellen. Diese Informationen werden im Allgemeinen in Form von Log-Daten abgelegt und dienen als Grundlage für die Analyse (BPA = Business Process Analysis) und das Monitoring (BAM = Business Activity Monitoring) der Prozesse.

Das Process Mining (auch: Workflow Mining) liefert hierzu entscheidende Beiträge. Während BPA und BAM a priori bekannte Geschäftsprozesse aufgreifen und analysieren, beschäftigt sich das Process Mining mit der Extraktion von Prozesswissen aus den Log-Daten der beteiligten Informationssysteme. Es geht dabei also nicht nur um die Analyse konkreter Prozessabläufe, sondern auch um die Erkennung kausaler Zusammenhänge zwischen den einzelnen Aktivitäten (engl. Tasks). Im Gegensatz zu den klassischen Ansätzen zur Prozessmodellierung, bei welchen die notwendigen Informationen typischerweise durch Mitarbeiterinterviews gewonnen werden, versucht das Process Mining die Geschäftsprozesse so zu modellieren, wie sie in der Realität ablaufen, und nicht wie sie (aufgrund der subjektiven Einschätzungen der Mitarbeiter) sein sollten.

Um dies zu erreichen müssen jedoch bestimmte Annahmen getroffen und Hindernisse überwunden werden.

Innerhalb des Process Mining wird grundlegend zwischen der Daten-, Organisations- und Prozesssicht unterschieden (siehe Kapitel 3.2). Während auf der Prozesssicht der Fokus der bisherigen Forschungs- und Entwicklungsanstrengungen lag, wurden die Daten- und die Organisationssicht kaum berücksichtigt¹. Ly begründet diesen Zustand damit, dass die Möglichkeiten, organisatorische Aspekte aus Log-Daten abzuleiten, sehr gering sind. Er führt weiter aus, dass Verlaufsdaten typischerweise nicht genügend Informationen für eine solche Aufgabe bieten². Demzufolge existiert heute eine Vielzahl von Ansätzen zur Ableitung von Prozessmodellen aus Log-Daten, während im Bereich der organisatorischen Aspekte verhältnismäßig wenige Arbeiten identifiziert werden können.

Andererseits kann im Allgemeinen davon ausgegangen werden, dass Informationen über die Aufbauorganisation eines Unternehmens a priori vorhanden sind oder unter geringem Aufwand beschafft werden können. Der Fokus der vorliegenden Arbeit liegt deshalb auf der Fragestellung, wie dieses Organisationswissen genutzt werden kann, um das Mining von Prozesswissen zu optimieren.

1.2 Gliederung der Arbeit

Um diese Fragestellung zu beantworten, soll die Themenfacette des Process Mining schrittweise vom Allgemeinen zum Speziellen untersucht werden, was auch im Aufbau der Arbeit zu erkennen ist. Zunächst wird auf die Grundlagen des Process Mining (Kapitel 3) eingegangen, wobei die Terminologie geklärt und ein Ordnungsrahmen zur Abgrenzung des Themenfokus erstellt wird. In Kapitel 4 geht es um die Verwendung von Organisationswissen im Process Mining. In diesem State-of-the-Art werden bisherige Ansätze identifiziert und vorgestellt und somit der aktuelle Forschungsstand herausgearbeitet.

Anschließend soll, auf Basis der Ergebnisse und neuer Ansätze, eine Methode zum Process Mining entwickelt werden, welche explizit vorhandenes Organisationswissen im Mining-Prozess verwendet (Kapitel 5). Diese Methode wird prototypisch in Form eines ProM-Plugins implementiert (Kapitel 6) und mit Echtdaten evaluiert (Kapitel 7). Die Evaluationsergebnisse werden dann zum Vergleich mit bisherigen Ansätzen verwendet.

¹ Song, M., van der Aalst, W.M.P.: Towards Comprehensive Support for Organizational Mining, Eindhoven University of Technology, Juni 2008, URL: <http://is.tn.tue.nl/staff/wvdaalst/publications/p484.pdf>, Abrufdatum: 10.04.2011.

² Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 57, Universität Ulm, URL: http://dbis.eprints.uni-ulm.de/543/1/Diplomarbeit_Ly05.pdf, Abrufdatum: 10.04.2011.

Abschließend werden die zentralen Ergebnisse der Arbeit zusammengefasst (Kapitel 8) und ein möglicher Ausblick auf diese Form des Process Mining angeboten (Kapitel 9).

2 Herangehensweise

Um die vorgestellten Zielsetzungen erreichen zu können, bedarf es des Einsatzes unterschiedlicher Forschungsmethoden.

Für die Grundlagen des Process Mining soll zunächst eine Querschnittsanalyse³ durchgeführt werden, bei welcher Inhaltsanalysen verschiedener allgemeiner und spezieller Arbeiten die Grundlage bilden. Eine anschließende quantitative Auswertung in Form einer Clusteranalyse soll die bisherigen Forschungsanstrengungen in den identifizierten Bereichen aufzeigen.

Beim State-of-the-Art zum Process Mining unter der Verwendung von Organisationswissen wird auf die Methode des Literatur-Reviews zurückgegriffen, wie sie durch Cooper⁴ beschrieben wird. Demnach bilden verschiedene Primäruntersuchungen die Grundlage des Reviews, wobei keine neuen primären Ergebnisse zur Forschungsfrage vorgestellt werden. Das Ziel des Reviews ist, die Ergebnisse ausgewählter Primäruntersuchungen zu bewerten, zusammenzufassen, zu beschreiben, zu klären oder zu integrieren. Das Review kann sich dabei auf inhaltliche, methodische, theoretische oder auch andere Eigenschaften der Primäruntersuchung stützen.⁵

Anschließend werden die Erkenntnisse des State-of-the-Art zur Entwicklung der neuen Mining-Methode herangezogen. Insbesondere sollen dabei unterschiedliche Verlaufsdaten und Organisationsmodelle intensiv beobachtet und analysiert werden, um induktiv zu einer gegenstandsverankerten Theoriebildung zu gelangen – man spricht dabei von Grounded Theory.⁶

Das Resultat soll dann prototypisch in Form eines ProM-Plugins implementiert werden (Prototyping), wodurch die Evaluation der Methode mit Beispiel- oder Echt Daten und ebenso der Vergleich mit bereits vorhandenen Methoden ermöglicht wird. Dazu ist ein iteratives Vorgehen mit den Phasen Theoriebildung, Implementierung und Evaluation durchaus sinnvoll und angedacht.

³ Wilder, T., Hess, T.: Forschungsmethoden der Wirtschaftsinformatik – Eine empirische Untersuchung, In: Wirtschaftsinformatik 49, S. 283.

⁴ Cooper, H.M.: Synthesizing Research – A Guide for Literature Reviews. 3. Aufl., Thousand Oaks et al. 1998.

⁵ Fettke, P.: State-of-the-Art des State-of-the-Art - Eine Untersuchung der Forschungsmethode „Review“ innerhalb der Wirtschaftsinformatik, Wirtschaftsinformatik 48 (2006) 4, S.258.

⁶ Wilder, T., Hess, T.: Forschungsmethoden der Wirtschaftsinformatik – Eine empirische Untersuchung, In: Wirtschaftsinformatik 49, S. 283.

3 Grundlagen des Process Mining

3.1 Begriffsklärung

Der Begriff Process Mining (auch: Workflow Mining, Control Flow Mining) hat seinen Ursprung im Data Mining. Dabei handelt es sich um einen Begriff mit populärwissenschaftlichem Ursprung, welcher sowohl in der Natur- als auch in den Wirtschaftswissenschaften verwendet wird.⁷

Nach Bissantz et. al. beschreibt Data Mining „die Extraktion implizit vorhandenen, nicht trivialen und nützlichen Wissens aus großen, dynamischen, relativ komplex strukturierten Datenbeständen. [...]“. Es wird versucht „Muster zu identifizieren, daraus Regeln abzuleiten, Unterschiede zwischen Gruppen von Datensätzen zu erkennen, diese Gruppen beschreibende Attribute zu bestimmen, die repräsentativsten Beispiele zu finden und Gleichungen zu konstruieren, die für numerische Variablen gelten“.⁸

Process Mining, als eine spezielle Form des Data Mining, zielt auf ein bestimmtes Wissen ab – das Prozesswissen⁹. Es wird also versucht, aus vorhandenen Datenbeständen (Verlaufsdaten) genau das Wissen abzuleiten, das für die Beschreibung eines Prozesses benötigt wird. In der Literatur werden diese Verlaufsdaten häufig auch *Audit Trail*, *History Data*, *Audit Data* oder *Event Data* genannt. Folgende Definitionen des Process Mining konnten weiterhin identifiziert werden:

- “The idea of process mining is to discover, monitor and improve real processes [...] by extracting knowledge from event logs.”¹⁰
- “The goal of a process discovery algorithm is to construct a process model which reflects the behavior that has been observed in the event log.”¹¹

⁷ H. Petersohn: Data Mining: Verfahren, Prozesse, Anwendungsarchitektur; Oldenbourg Wissenschaftsverlag 2005, S. 8.

⁸ N. Bissantz; J. Hagedorn: Data Mining (Datenmustererkennung) , In: Wirtschaftsinformatik Volume 35 (1993), S. 481.

⁹ Van der Aalst, W: Process Mining, In: Liu, Ling; Özsu, M.T.: Encyclopedia of Database Systems, Springer 2009, S. 2173.

¹⁰ Jans, M., Lybaert, N., Vanhoof, K: Business Process Mining for Internal Fraud Risk Reduction: Result of a Case Study, URL: <http://www.ecmlpkdd2008.org/sites/ecmlpkdd2008.org/files/pdf/workshops/ipm/3.pdf>, Abrufdatum: 17.07.2011.

¹¹ Rozinat, A., de Medeiros, A.K., Günther, C.W., Weijters, A.J.M.M., van der Aalst, W.M.P.: Towards an Evaluation Framework for Process Mining Algorithms, In: BPM'07 Proceedings of the 2007 international conference on Business process management.

Beide Definitionen sind sehr eng gefasst, da sie sich nur auf eine bestimmte Sicht des Process Mining – die Prozesssicht – beziehen. Die verschiedenen Sichten werden in Kapitel 3.2 vorgestellt.

Vereinfacht lässt sich das Vorgehen im Process Mining wie folgt abbilden.

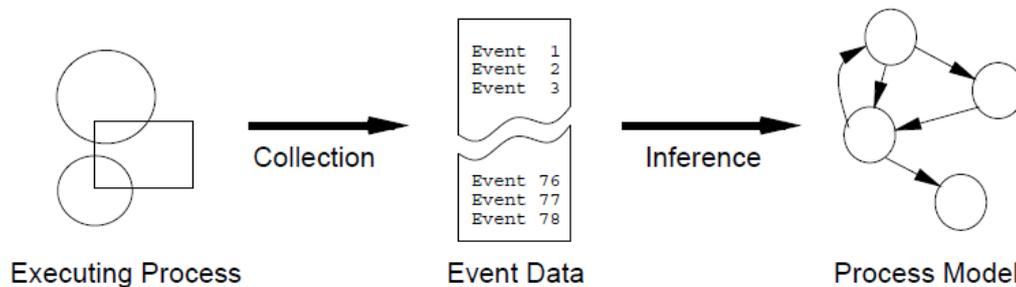


Abb. 1: Informationsquelle des Process Mining¹²

Bissantz et. al¹³ und Peterson¹⁴ beschreiben darüber hinaus typische Charakteristiken der Verfahren zum Data Mining, welche ebenfalls auf die Verfahren zum Process Mining übertragen werden können:

- „limately understandable“: Die gewonnen Informationen werden in verständlicher Form (z.B. Text oder Grafik) präsentiert oder von einer Software weiterverarbeitet. Im Falle des Process Mining handelt es sich dabei meist um grafische Prozessmodelle.
- „valid“: Die generierten Thesen sind mit einer statistischen Sicherheit versehen. Demnach sollen die resultierenden Prozessinformationen die Basis-Loginformationen repräsentieren.
- „novel“: Der Fokus liegt auf neuen Erkenntnissen – detaillierte Informationen sind im Kapitel 3.3 zu finden.
- „non-trivial“: Die ermittelten Informationen sind nicht trivial.
- „potential useful“: Die gewonnenen Information sind nicht immer offensichtlich bewertbar, aber nützlich.
- „runtime“: Die Laufzeit des Mining-Prozesses liegt in einem vertretbaren Rahmen und sollte in Abhängigkeit der Anzahl der zu untersuchenden Datensätze nicht stärker als ein Polynom niedrigen Grades steigen.

¹² Cook, J.E., Wolf, A.L.: Automating Process Discovery through Event-Data Analysis, In: ICSE '95 Proceedings of the 17th international conference on Software engineering, S. 74.

¹³ N. Bissantz; J. Hagedorn: Data Mining (Datenmustererkennung) , In: Wirtschaftsinformatik Volume 35 (1993), S. 481.

¹⁴ H. Petersohn: Data Mining: Verfahren, Prozesse, Anwendungsarchitektur; Oldenbourg Wissenschaftsverlag 2005, S. 9.

Diese Charakteristiken werden auch bei der Entwicklung der Mining-Methode unter Verwendung von Organisationswissen eine Rolle spielen.

3.2 Sichten des Process Mining

Wie bereits einleitend ausgeführt, wird beim Process Mining zwischen drei grundlegenden Sichten unterschieden - der Prozesssicht, der Organisationssicht und der Fallsicht.¹⁵

Bei der Prozesssicht liegt der Fokus auf dem Kontrollfluss, also beispielsweise auf der Ausführungsreihenfolge von Aktivitäten. Das Ziel dieser Sicht liegt in der Erfassung eines Modells, welches alle möglichen Prozesspfade bestmöglich abbildet. Dies wird im Allgemeinen mit Petri-Netzen, ereignisgesteuerten Prozessketten (= EPK), endlichen Zustandsautomaten (engl. Finite State Machine = FSM) oder gerichteten Graphen realisiert. Es ist also festzustellen, dass in den bisherigen Ansätzen keine Einigkeit über das Output-Format existiert, was einen Vergleich der Methoden erschwert. Zwar können Petri-Netze und EPKs transformiert werden, jedoch nur unter bestimmten Bedingungen. Bei Zustandsautomaten gestaltet sich diese Transformation bedeutend schwieriger. Erstaunlich ist dabei weiterhin, dass die Mehrzahl der Ansätze ihre Ergebnisse als Petri-Netz visualisiert¹⁶, obwohl es sich dabei um eine verhältnismäßig alte und eher unanschauliche Notation handelt.

Die Organisationssicht (auch: Organizational Mining) beschäftigt sich mit den innerbetrieblichen Organisationsstrukturen. Es wird untersucht, welche Akteure an einem Prozess beteiligt sind, und in welcher Beziehung diese zueinander stehen. Das Ziel liegt dabei in der Ableitung einer Organisationsstruktur, indem den Akteuren Rollen und Organisationseinheiten zugeordnet und diese miteinander in Verbindung gesetzt werden. Beispielsweise kann hierdurch eine Art soziales Netzwerk abgebildet werden. Als Output der entsprechenden Mining-Methoden kommen in der Regel Soziogramme oder Organigramme zum Einsatz. Bei genauerer Betrachtung dieser Visualisierungsformen ist allerdings anzumerken, dass, im Gegensatz zur Prozesssicht, unterschiedliche Informationen abgebildet werden. Während Soziogramme die Beziehungen zwischen den Prozessbeteiligten darstellen bilden Organigramme die Aufbauorganisation eines Unternehmens ab.

¹⁵ Van der Aalst, W.M.P.: Business Alignment – Using Process mining as a Tool for Delta Analysis and Conformance Testing, In: Requirements Engineering, Volume 10, Springer 2005, S. 1998ff .

¹⁶ Tiwari, A., Turner, C.J.: A review of Business Process Mining – State-of-the-Art and Future Trends, In: Business Process Management Journal, Vol. 14, S. 7.

Die Fallsicht legt den Fokus auf die Eigenschaften einzelner Prozessinstanzen. Fälle (engl. Cases) können demnach nicht nur über Prozesspfade und die zugehörigen Akteure charakterisiert werden, sondern auch über fallspezifische Werte, wie beispielsweise die Anzahl der Produkte in einer Bestellung. Eine konkrete Anwendung dazu stellt z.B. das Decision Mining¹⁷ dar, bei welchem analysiert wird, wie Daten-Attribute die Entscheidungen in aktuellen Prozessen basierend auf vergangenen Prozess-Ausführungen beeinflussen. Es handelt sich also um eine Entscheidungspunkt-Analyse, welche nach Abhängigkeiten in Daten sucht, die den Verlauf einer Prozessinstanz beeinflussen.

Darüber kann die Datensicht identifiziert werden, welche sich mit der Extraktion eines Datenmodells aus den Log-Daten befasst. Im Gegensatz zur Fallsicht (konkret zum Decision Mining) werden also nicht die Auswirkungen von Datenattributen auf die Ausführung betrachtet, sondern die Strukturierung der Daten. Bisher sind in diesem Bereich jedoch keine Arbeiten bekannt.

Weiterhin konnte in der Literatur die Anwendungssicht¹⁸ identifiziert werden. Der Fokus liegt dabei auf den notwendigen (Software-)Anwendungen, die für die Prozessausführung genutzt werden. Dieser Sicht kommt jedoch, gerade auch aufgrund der heute sehr umfangreichen BPM-Suiten, eine sehr geringe Bedeutung zu.

Über die genannten Sichten hinaus, sind auch Mischformen oder Abwandlungen denkbar. Beispielsweise entwickelte Ly ein Verfahren namens „Staff Assignment Mining“, um Mitarbeiterzuordnungsregeln aus Verlaufsdaten und einem a priori vorhandenen Organisationsmodell abzuleiten.¹⁹

An dieser Stelle sei angemerkt, dass die vorliegende Arbeit und die zu entwickelnde Mining Methode im Wesentlichen der Prozesssicht zuzuordnen ist, da das Ziel in der Entwicklung eines Prozessmodells liegt. Für das Mining sollen jedoch über die Log-Daten hinaus Informationen über die Aufbaustruktur des Unternehmens herangezogen werden.

¹⁷ Rozinat, A., van der Aalst, W.M.P.: Decision Mining in Business Processes, Eindhoven University of Technology, März 2006, URL: <http://www.wis.win.tue.nl/~wvdaalst/publications/p312.pdf>, Abrufdatum: 10.05.2011.

¹⁸ Van der Aalst, W.M.P., Weijters, A.J.M.M.: Process Mining – A Research Agenda, Eindhoven University of Technology, Juli 2003, URL: <http://tmitwww.tm.tue.nl/staff/wvdaalst/Publications/p219.pdf>, Abrufdatum: 10.05.2011.

¹⁹ Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 57, Universität Ulm, URL: http://dbis.eprints.uni-ulm.de/543/1/Diplomarbeit_Ly05.pdf, Abrufdatum: 10.04.2011, S. 63ff.

3.3 Einsatzmöglichkeiten

Die Grundidee des Process Mining besteht in der Identifizierung, dem Monitoring und der Optimierung von realen Prozessen durch Extraktion von Wissen aus Ereignisprotokollen.²⁰ Diese unterschiedlichen Einsatzmöglichkeiten sollen im Folgenden erläutert werden.

Der populärste und am weitesten verbreitete Ansatz liegt in der Identifikation der Ablaufstrukturen eines Unternehmens. Bisher führte der Weg zu einem Prozessmodell typischerweise über die Durchführung von Interviews, bei welchen Mitarbeiter zu Ihren „eigenen“ Prozessen befragt werden. Modelle, die mit Process Mining-Technologien erstellt werden, sind, anders als Modelle, die mit klassischen Techniken entwickelt werden, objektiv und reflektieren somit den tatsächlichen Prozessablauf, wobei gleichzeitig erhebliche Zeit- und Kosteneinsparungen zu verbuchen sind²¹. Man bezeichnet dieses Vorgehen zur reinen Prozessidentifikation als „Process Discovery“.

Sollte ein Prozessmodell a priori vorhanden sein, sind mehrere Szenarien denkbar. Das durch Process Mining entwickelte Modell kann zum Vergleich mit dem bereits vorhandenen Prozessmodell herangezogen werden. Dadurch können Differenzen zwischen Ist- und Soll-Zustand identifiziert und analysiert werden²² – bei diesem Vorgehen spricht man von der Delta-Analyse²³. Die Einsatzmöglichkeit wird im Allgemeinen als „Conformance“ (dt.: Übereinstimmung) bezeichnet. Die „Conformance“ kann weiterhin durch Organisationsmodelle, Business Rules, Qualitätsanforderungen (Quality of Service = QoS) und vieles mehr geprüft werden.²⁴

Grobe und häufige Abweichungen des Ist-Zustandes vom Soll-Zustand stellen jedoch die Korrektheit und Angemessenheit des a priori vorhandenen Modells in Frage. Dadurch kann es sinnvoll sein, das vorhandene Modell auf Basis der Erkenntnisse aus der Delta-Analyse und somit an die realen Strukturen und Anforderungen anzugleichen. Demnach kann Process Mining ebenfalls als eine

²⁰ Van der Aalst, W.M.P.: Process Mining, In: Liu, Ling; Özsu, M.T.: Encyclopedia of Database Systems, Springer 2009, S. 2171.

²¹ Braun, F., Kuhn, R.: Der Einsatz von Process Mining Techniken in Revision und Compliance, Deutsches Institut für Interne Revision e.V. / ProcessGold AG, <http://www.slideshare.net/ProcessGold/der-einsatz-von-process-mining-techniken-in-revision-und-compliance>, Abrufdatum: 12.05.2011.

²² Van der Aalst, W.M.P.: Decision Support Based on Process Mining, In: Burstein, F; Holsapple C.W.: Handbook on Decision Support Systems 1 – Basic Themes, International Handbooks on Information Systems 2008, S. 637.

²³ Van der Aalst, W.M.P.: Business Alignment: Using Process Mining as a Tool for Delta Analysis, In: CAiSE Workshops (2), 2004, S. 144.

²⁴ Van der Aalst, W.M.P.: Process Mining, In: Liu, Ling; Özsu, M.T.: Encyclopedia of Database Systems, Springer 2009, S. 2172.

Methode zur Verbesserung der Prozessqualität durch Ableitung von Prozesswissen verstanden werden.²⁵

Die gewonnenen Prozessinformationen können außerdem zur Erweiterung des Vergleichsmodells genutzt werden. Dabei wird, ohne die vorherige Überprüfung der „Conformance“ (also ohne Delta-Analyse), eine Anreicherung des vorhandenen Prozessmodells mit neuen Informationen, wie beispielsweise Performance-Daten, durchgeführt²⁶. Dieses Vorgehen wird als „Extension“ bezeichnet.

Diese unterschiedlichen Einsatzmöglichkeiten des Process Mining werden von van der Aalst wie folgt visualisiert.

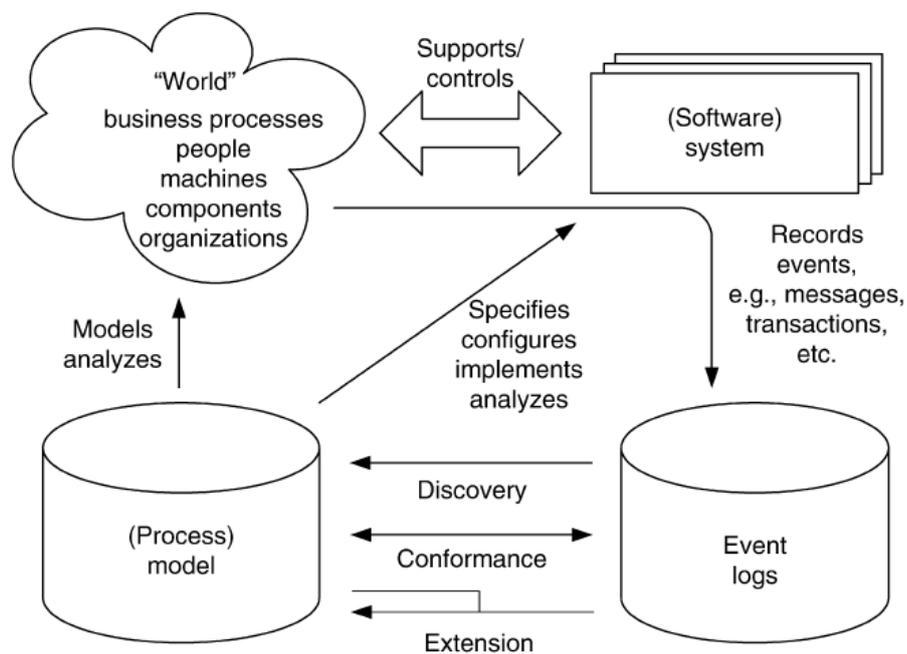


Abb. 2: Three types of process mining: (i) discovery, (ii) conformance, and (iii) extension²³

Die Abbildung zeigt das Zusammenwirken der unterschiedlichen Komponenten und wie dadurch die Zielsetzungen umgesetzt werden. Kritisch ist jedoch anzumerken, dass in der Abbildung kein Pfad existiert, welcher auf Basis der Conformance zu einem neuen Prozessmodell führt („Adaption“).

²⁵ Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 7, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

²⁶ Van der Aalst, W.M.P.: Process Mining, In: Liu, Ling; Özsu, M.T.: Encyclopedia of Database Systems, Springer 2009, S. 2172.

3.4 Anforderungen an Verlaufsdaten

Um ein Process Mining, unabhängig von Sichtweise und Einsatzzweck, durchführen zu können, werden Informationen über die einzelnen Prozessschritte benötigt. Wie bereits einleitend angesprochen, werden diese Protokoll-Informationen von den beteiligten Informationssystemen in Form von Log-Daten (hier: Event Logs²⁷) zur Verfügung gestellt. Log-Daten müssen bestimmten Anforderungen genügen, damit Process Mining-Techniken Wissen extrahieren können. Diese Anforderungen werden in der Literatur nahezu durchgängig wie folgt formuliert:²⁸

1. Jedes Ereignis verweist auf eine Aktivität (engl. Task)
2. Jedes Ereignis verweist auf eine Prozessinstanz (engl. Case)
3. Jedes Ereignis hat einen Initiator auf welchen verwiesen wird
4. Ereignisse haben einen Zeitstempel und sind sortiert

Tabelle 1 stellt ein solches Event-Log exemplarisch dar.

| Case | Task | Initiator | Timestamp |
|------|------|-----------|------------------|
| 1 | A | Manfred | 9-05-2011 15:01 |
| 2 | A | Manfred | 9-05-2011 15:12 |
| 3 | A | Susanne | 9-05-2011 16:03 |
| 3 | D | Inge | 9-05-2011 16:07 |
| 1 | B | Mike | 9-05-2011 18:25 |
| 1 | H | Manfred | 10-05-2011 09:23 |
| 2 | C | Mike | 10-05-2011 10:34 |
| 4 | A | Susanne | 10-05-2011 10:35 |
| 2 | H | Manfred | 10-05-2011 12:34 |
| 3 | E | Peter | 10-05-2011 12:50 |
| 3 | F | Inge | 11-05-2011 10:12 |
| 4 | D | Peter | 11-05-2011 10:14 |
| 3 | G | Susanne | 11-05-2011 10:44 |
| 3 | H | Peter | 11-05-2011 11:03 |
| 4 | F | Susanne | 11-05-2011 11:18 |
| 4 | E | Tim | 11-05-2011 12:22 |

Tabelle 1: Event-Log²⁹

²⁷ Van der Aalst, W.M.P.: Business Alignment: Using Process Mining as a Tool for Delta Analysis, In: CAiSE Workshops (2), 2004, Abstract.

²⁸ Van der Aalst, W.M.P.: Decision Support Based on Process Mining, In: Burstein, F; Holsapple C.W.: Handbook on Decision Support Systems 1 – Basic Themes, International Handbooks on Information Systems 2008, S. 640.

²⁹ Eigene Erstellung in Anlehnung an: Van der Aalst, W.M.P.: Decision Support Based on Process Mining, In: Burstein, F; Holsapple C.W.: Handbook on Decision Support Systems 1 – Basic Themes, International Handbooks on Information Systems 2008, S. 641.

Bei näherer Betrachtung entsteht jedoch Diskussionsbedarf bzgl. dieser Anforderungen, da eventuell Einschränkungen geschaffen werden, die nicht unbedingt erforderlich sind oder nur in bestimmten Problemstellungen Relevanz finden.

Die genannten Anforderungen sind die Grundlage von nahezu allen gängigen Ansätzen zum Process Mining, allerdings wird bei der Betrachtung der konkreten Techniken deutlich, dass bereits vor dem Mining-Prozess einige dieser Informationen verworfen werden. Sowohl algorithmische Verfahren, wie der α -Algorithmus oder Ktail, also auch statistische (z.B. RNet) und hybride (z.B. Markov) Verfahren benötigen lediglich eine geordnete Abfolge von Aktivitäten³⁰. Unter Ordnung ist dabei eine Sortierung nach Case in erster Priorität, und Zeitstempel in zweiter Priorität zu verstehen. Passend zum Event-Log ergäbe sich folgende Aktivitätsfolge.

| |
|------------------------|
| ABHACHADEFKADFE |
|------------------------|

Diese Folge enthält schließlich auch nur noch Teile der ursprünglich geforderten Informationen, da beispielsweise nicht mehr erkennbar ist, mit welcher Aktivität ein Case startet oder abschließt. Die Zuordnung einer konkreten Aktivität zu einem Case kann ebenfalls nicht mehr erfolgen. Es muss allerdings festgehalten werden, dass Case und Zeitstempel für die korrekte Sortierung der Folge notwendig bleiben. Informationen über den Initiator werden hingegen an keiner Stelle verarbeitet und demnach auch nicht benötigt.

Diese Information wird allerdings bei der Lösung anderer Problemstellungen, wie beispielsweise der Erstellung sozialer Netzwerke im Bereich des Organizational Mining notwendig. Auch in der vorliegenden Arbeit wird diese Information eine wichtige Rolle spielen, da es sich um die Schnittstelle der Log-Daten zum Organisationswissen handelt.

Die Ausführungen zeigen, dass es nicht unbedingt erforderlich ist, die Anforderungen an Verlaufsdaten für das Process Mining global zu definieren, sondern diese im Hinblick auf die konkrete Mining-Methode zu erheben.

³⁰ Cook, J.E., Wolf, A.L.: Discovering Models of Software Processes from Event-Based Data, In: ACM Transactions on Software Engineering and Methodology – Journal, Vol. 7, Issue 3, S. 225ff.

3.5 Phasen im Process Mining

3.5.1 Pre-Processing

Die erste der drei Phasen im Process Mining befasst sich mit der Extraktion der relevanten Informationen aus den Verlaufsdaten. Da diese Verlaufsdaten zu einem bestimmten Zeitpunkt aus Informationssystemen exportiert werden, ist es wahrscheinlich, dass darin nicht vollständig ausgeführte Prozessinstanzen vorhanden sind. Der Zeitraum, welcher für den Export gewählt wurde, startet also entweder nach der Initiierung eines solchen Prozessen, oder endet vor dem Abschluss der Prozessinstanz. Weiterhin können Prozessinstanzen enthalten sein, welche aufgrund von Fehlern abgebrochen und somit nicht vollständig sind. Diese Daten würden in der anschließenden Processing-Phase unerwünschte Effekte verursachen, da Prozesse analysiert würden, welche in dieser Form nicht existieren. Aus diesem Grund wird versucht, die betroffenen Log-Einträge im Vorfeld zu entfernen.

Da es sich beim Process Mining um ein induktives Lernen³¹ aus Beispielen handelt, kann zwischen dem Lernen aus positiven und negativen Beispielen unterschieden werden³². Bei positiven Beispielen handelt es sich um korrekt ausgeführte Prozessinstanzen, während bei negativen Beispielen fehlerhafte Ausführungen oder Abbrüche betrachtet werden. Es wäre also auch denkbar, statt der korrekten Instanzen, die fehlerhaften Instanzen in der Phase des Pre-Processing zu selektieren, allerdings gestaltet sich die Unterscheidung in der Praxis als zu kompliziert, da die notwendigen Informationen in den Verlaufsdaten nur selten vorhanden sind. Deshalb, und da alle aktuellen Ansätzen mit positiven Beispielen arbeiten, wird im Pre-Processing versucht, die fehlerhaften Informationen zu entfernen.

Hierbei ist festzustellen, dass aufgrund der genannten Schwierigkeit im Allgemeinen keine vollständig korrekten Verlaufsdaten extrahiert werden können. Daraus resultieren unterschiedliche Problemstellungen (siehe Kapitel 3.6), welche durch die Mining-Methode bearbeitet werden müssen.

3.5.2 Processing

Das Processing stellt die Kern-Phase des Process Mining dar, in welcher der eigentliche Mining-Algorithmus zur Anwendung kommt. Im Allgemeinen un-

³¹ Herbst, J.: Ein induktiver Ansatz zur Akquisition und Adaption von Workflow-Modellen, Tenea Verlag für Media, Berlin 2004, S. 8ff.

³² Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 23f, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

terscheidet man dabei zwischen algorithmischen, statistischen und hybriden Ansätzen, auf welche in Kapitel 3.7 genauer eingegangen wird. In dieser Phase ist der aktuelle Forschungsschwerpunkt des Process Mining, unabhängig von der Sichtweise, zu sehen.

Das Ergebnis des Pre-Processing dient dieser Phase als Input – darauf wird der Mining-Algorithmus angewendet, welcher einerseits versucht Prozesswissen abzuleiten und andererseits dieses Wissen in Form eines (Prozess-)Modells grafisch zu visualisieren. Bei den Modellen wird zwischen vollständigen Prozessmodellen, die das gesamte Log repräsentieren, und der Repräsentation häufig ausgeführter Prozessmuster unterschieden, was als *Workflow Pattern Mining* bezeichnet wird.³³ Für weiterführende Informationen dazu sei auf die Arbeit von Russell et. al.³⁴ verwiesen. Generell sollte das Modell möglichst konsistent zu den Verlaufsdaten sein. Da ein Prozess jedoch auf unterschiedliche Weise modelliert werden kann, sollte das aus dem Processing resultierende Modell weiterhin möglichst minimal sein. Unterschiedliche Ansätze realisieren das durch einen abschließenden Schritt im Mining-Prozess. Beispielhaft wird dazu auf die Arbeit von Cook et. al.³⁵ verwiesen. Andere Ansätze sehen diesen Schritt auch in der Phase des Post-Processing³⁶.

3.5.3 Post-Processing

Nach der Erstellung des Prozessmodells im Processing kann, zusammen mit den ursprünglichen Verlaufsdaten, eine Anreicherung dieses Modells mit zusätzlichen Informationen vorgenommen werden. Durch die Simulation der in den Log-Daten vorhandenen Prozessinstanzen können beispielsweise statistische Werte wie Häufigkeiten oder Wahrscheinlichkeiten berechnet und in das Modell integriert werden.

³³ Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 16f, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

³⁴ Russell, N., ter Hofstede, A.H.M., van der Aalst, W.M.P, Mulyar, N.: Workflow Control-Flow Patterns – A Revised View, <http://www.workflowpatterns.com/documentation/documents/BPM-06-22.pdf>, Abrufdatum: 12.05.2011.

³⁵ Cook, J.E.; Wolf, A.L.: Discovering Models of Software Processes from Event-Based Data, In: ACM Transactions on Software Engineering and Methodology, Volume 7 Issue 3, Juli 1998, S. 234.

³⁶ de Medeiros, A.K.A, van Dongen, B.F., van der Aalst, W.M.P, Weijters, A.J.M.M.: Process Mining – Extending the α -Algorithm to Mine Short Loops, <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.5.7518&rep=rep1&type=pdf>, Abrufdatum: 12.05.2011, S. 2f.

3.6 Problemstellungen

Wie bereits in Kapitel 3.5.1 angedeutet, existieren mehrere Problemstellungen, welche durch die Mining-Techniken gelöst werden sollten. Diese Problemstellungen ergeben sich einerseits aus den Verlaufsdaten direkt, da sie fehlerbehaftet sein können, als auch indirekt, da sie korrekt interpretiert werden müssen. Im Folgenden sollen die möglichen Problemstellungen erläutert werden.

- *Rauschdaten (engl. Noise)*: Informationen können während der Protokollierung verloren gehen oder aufgrund von Verzögerungen an einer falschen Stelle erscheinen. Ebenfalls könnten für die Prozessausführung irrelevante Ereignisse, wie beispielsweise Telefonate, mitprotokolliert werden. Diese Situationen verursachen fehlerhafte Kausalbeziehungen und somit fehlerhafte Relationen, welche wiederum das Ergebnis verfälschen.³⁷
- *Versteckte Tasks (engl. Hidden Tasks)*: Werden Prozessschritte manuell ausgeführt, können diese häufig nicht protokolliert werden. Gleiches gilt für Prozesspfade, die nicht, oder nur selten ausgeführt werden.³⁸ Da nur im Log sichtbare Tasks aufgedeckt werden können, besteht das Problem, dass diese unberücksichtigt bleiben.
- *Doppelte Tasks (engl. Duplicate Tasks)*: Hierbei handelt es sich um Situationen, in denen eine Aktivität unterschiedlichen Knoten im Prozessmodell zugeordnet wird. In diesem Fall kann nicht erkannt werden, dass es sich um die gleiche Aktivität handelt.³⁹
- *Parallele / konkurrierende Tasks (Concurrent Tasks)*: In manchen Prozesssequenzen spielt die Reihenfolge der Ausführung keine Rolle, da die enthaltenen Aktivitäten parallel ablaufen können. Es ist allerdings nicht immer trivial diese Parallelität zu erkennen, wenn beispielsweise die Ausführungsreihenfolge der Aktivitäten in den betrachteten Prozessinstanzen nicht variiert.
- *Nicht-wahlfreie Konstrukte (engl. Non-Free-Choice)*: Innerhalb eines Prozesses können Abhängigkeiten zwischen entfernten Aktivitäten existieren. Einige Algorithmen, wie beispielsweise der α -Algorithmus,

³⁷ Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 22,

<http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

³⁸ Tiwari, A., Turner, C.J.: A review of Business Process Mining – State-of-the-Art and Future Trends, In: Business Process Management Journal, Vol. 14, S. 12.

³⁹ Van der Aalst, W.M.P., Weijters, A.J.M.M.: Process Mining – A Research Agenda, Eindhoven University of Technology, Juli 2003, URL:

<http://tmitwww.tm.tue.nl/staff/wvdaalst/Publications/p219.pdf>, Abrufdatum: 10.05.2011, S.7.

betrachten jede Aktivität atomar, sodass diese Abhängigkeiten verloren gehen. Dies wird als „non-free-choice“-Problem bezeichnet.⁴⁰

- *Schleifen (engl. Loops)*: Innerhalb eines Prozesses besteht die Möglichkeit, dass eine Aktivität oder eine Sequenz von Aktivitäten, mehrfach ausgeführt wird, was typischerweise als Schleife bezeichnet wird. Gehört dabei nur eine Aktivität zur Schleife, ist dies noch einfach zu erkennen, da diese unterschiedlich oft nacheinander in den Verlaufsdaten erkennbar ist. Komplizierter wird dies jedoch bei Sequenzen, da dadurch zu jedem Punkt im Prozess zurückgesprungen werden kann.⁴¹
- *Ausführungsfehler und Ausnahmen*: Während der Ausführung von Prozessen können Fehler, sowohl menschlicher als auch technischer Natur, auftreten, wodurch der Prozess sein Ziel verfehlt oder ineffizient wird. Im Gegensatz zu Rauschdaten tritt hier also ein Fehler während der Ausführung und nicht während der Protokollierung auf.⁴²
- *Datenerfassung verschiedenartiger Quellen (heterogeneous sources)*: Die Verlaufsdaten, welche zum Process Mining herangezogen werden, stammen im Allgemeinen aus verschiedenen Informationssystemen (siehe Kapitel 1). Dabei ist es unwahrscheinlich, dass alle Systeme die Log-Daten auf identische Weise zur Verfügung stellen, sodass Lösungen gefunden werden müssen, diese Daten anzugleichen.⁴³

Darüber hinaus wird in der Literatur häufig auch die Herausforderung der Ergebnisvisualisierung angesprochen. Beispielsweise sollte das resultierende Prozessmodell in einer leicht verständlichen Form dargestellt werden. Das ist im Allgemeinen problematisch, da die extrahierten Verlaufsdaten aus den beteiligten Informationssystemen nicht nur einen, sondern meist mehrere Prozesse repräsentieren. Im Idealfall kann eine Mining-Methode diese Prozesse voneinander trennen (engl.: „clustering“), sodass mehrere Modelle erzeugt werden.

Aktuelle kommerzielle Produkte, wie beispielsweise der ARIS PPM (= „Process Performance Manager“) legen den Fokus daher eher auf die Darstellung

⁴⁰ Sun, J; Wand, J; Wen, L, van der Aalst, W.M.P.: Mining process models with non-free-choice constructs, In: Data Mining and Knowledge Discovery, Volume 15 (2007), S. 156f.

⁴¹ Van der Aalst, W.M.P., Weijters, A.J.M.M.: Process Mining – A Research Agenda, Eindhoven University of Technology, Juli 2003, URL: <http://tmitwww.tm.tue.nl/staff/wvdaalst/Publications/p219.pdf>, Abrufdatum: 10.05.2011, S.8f.

⁴² Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 23, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

⁴³ Van der Aalst, W.M.P., Weijters, A.J.M.M.: Process Mining – A Research Agenda, Eindhoven University of Technology, Juli 2003, URL: <http://tmitwww.tm.tue.nl/staff/wvdaalst/Publications/p219.pdf>, Abrufdatum: 10.05.2011, S. 11f.

von Performance-Indikatoren wie Ausführungszeiten und Ähnliches, und nicht auf die Darstellung von Prozessmodellen.

Weiterhin wird von der Herausforderung der Delta-Analyse gesprochen, welche sich mit dem Vergleich des a-priori Modells mit dem neuen Modell beschäftigt. Die Durchführung dieses Vergleichs ist deshalb nicht trivial, weil (a) ein Prozessmodell unterschiedlich modelliert werden kann und (b) sich die Modellierungssprachen der beiden Modelle unterscheiden können. Im Fall (a) muss nach genauer Betrachtung der Modelle entschieden werden, ob eine Abweichung von Ist- und Sollzustand existiert, im Fall (b) könnte eine Modelltransformation einen Lösungsansatz darstellen, welcher wiederum meist zu Fall (a) führen wird.

3.7 Mining-Methoden

Prinzipiell können die bisherigen Methoden zum Process Mining in drei methodische Kategorien unterteilt werden: algorithmische, statistische und hybride Verfahren⁴⁴.

Algorithmische Verfahren konstruieren ein Prozessmodell auf Basis von Ordnungsrelationen. Als Beispiel für die algorithmischen Verfahren kann der α -Algorithmus genannt werden, dessen Ordnungsrelationen immer genau zwei Ereignistypen betrachten. Andere Algorithmen in diesem Segment, wie beispielsweise Ktail, betrachten auch ganze Ereignissequenzen, wodurch auch entfernte Abhängigkeiten aufgedeckt werden können.

Statistische Verfahren, wie beispielsweise RNet, auch neuronale Netzwerke genannt, konstruieren ein Prozessmodell hingegen auf der Grundlage von Häufigkeiten bzw. Wahrscheinlichkeiten von Ereignissequenzen. Aus diesem Grund können statistische Verfahren im Allgemeinen besser mit Rauschdaten umgehen, da diese verhältnismäßig selten auftreten und durch eine bestimmte Parametrisierung der Algorithmen ausgeblendet werden können. Problematisch ist hierbei allerdings, dass im Falle der Ausblendung von Informationen die Konsistenz zu den Log-Daten nicht mehr garantiert werden kann.

Die hybriden Verfahren, wie beispielsweise Markov, vereinen algorithmische und statistische Verfahren. Häufig wird das Grundmodell algorithmisch erstellt, anschließend wird die Behandlung von Rauschdaten durch statistische Verfahren realisiert.

⁴⁴ Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 26, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

Über diese Verfahren hinaus können aufgrund aktueller Entwicklungen semantische Ansätze abgegrenzt werden. Seit 2008 können in diesem Bereich vereinzelt Arbeiten, beispielsweise von de Medeiros und van der Aalst⁴⁵, identifiziert werden, welche sich mit der semantischen Beschreibung von Log-Informationen, sowie ersten semantischen Mining-Ansätzen auseinandersetzen (weitere Informationen über die bisherige Literatur zum Process Mining sind in Kapitel 3.9 zu finden). Es ist davon auszugehen, dass entsprechende Verfahren, gerade im Hinblick auf aktuelle Themen wie dem Semantic Web, an Bedeutung gewinnen werden.

Im Bereich der algorithmischen Verfahren könnte weiterhin eine spezielle Betrachtung von genetischen Algorithmen vorgenommen werden, da diese in den Entwicklungen der letzten Jahre stark an Bedeutung gewonnen haben⁴⁶.

3.8 Ordnungsrahmen zum Process Mining

Auf Basis der bisherigen Ausführungen wird nun zusammenfassend ein tabellarischer Ordnungsrahmen erstellt, um einen aggregierten Überblick über das Process Mining anzubieten.

| | | | | |
|----------------------|---------------------------------|-------------|---|------------------------|
| Sichten | Prozesssicht | | Organisationssicht | Fallsicht |
| | Datensicht | | | Anwendungssicht |
| Einsatzmöglichkeiten | Process Discovery | | | Conformance |
| | Process Adaption | | | Extension |
| Phasen | Pre-Processing | | Processing | Post-Processing |
| Methoden | algorithmisch | statistisch | hybrid | semantisch |
| Problemstellungen | Rauschdaten | | Versteckte Tasks | Doppelte Tasks |
| | Ausführungsfehler und Ausnahmen | | Nicht-wahlfreie Konstrukte | Schleifen |
| | Parallele/konkurrierende Tasks | | Datenerfassung verschiedenartiger Quellen | Ergebnisvisualisierung |

Tabelle 2: Ordnungsrahmen zum Process Mining

⁴⁵ de Medeiros, A.K.A, van der Aalst, W.M.P.: Process Mining towards Semantics, In: Advances in Web Semantics.

⁴⁶ Tiwari, A., Turner, C.J.: A review of Business Process Mining – State-of-the-Art and Future Trends, In: Business Process Management Journal, Vol. 14, S. 7ff.

Auffällig an dem Ordnungsrahmen ist, dass im Bereich der Problemstellung das Argument der Delta-Analyse nicht erfasst wurde. Dies ist im Wesentlichen auf zwei Gründe zurückzuführen:

1. Im Gegensatz zu den anderen Problemstellungen ist dieses Argument abhängig vom Einsatzzweck (tritt nur bei Conformance und Adaption auf).
2. Es ist keine eindeutige Lösung für das Problem möglich, da diese wesentlich vom a-priori Modell abhängen würde.

3.9 Literaturübersicht und aktuelle Entwicklungen

In die Themenfacette des Process Mining wird aktuell sehr viel Forschungsarbeit investiert, sodass es an dieser Stelle sicherlich interessant ist, die aktuellen Entwicklungen und den Stand der Forschung aufzuzeigen. Insbesondere sollen die bisherigen Arbeiten zentral und mit einigen Eckinformationen genannt und analysiert werden.

Es sei dazu angemerkt, dass es sich bei diesen Arbeiten ausschließlich um wissenschaftliche Papiere handelt, die Vollständigkeit aufgrund der Masse jedoch nicht gewährleistet werden kann. Die Identifizierung der Arbeiten erfolgte einerseits auf der Basis relevanter Literaturverzeichnisse und andererseits durch Stichwortsuchen innerhalb der Literaturdatenbanken Springerlink.com und Google Scholar.

Im Rahmen dieser Recherche wurden insgesamt 78 Arbeiten identifiziert, welche sich ausschließlich mit dem Process Mining befassen. Diese Arbeiten wurden in drei Kategorien eingeordnet:

| Kategorie | Anzahl der Arbeiten |
|---------------|---------------------|
| Methode | 65 |
| Überblick | 6 |
| Weiterführend | 7 |

Tabelle 3: Literaturkategorien und Anzahl bisheriger Arbeiten

Bei den „Methoden“ handelt es sich um Arbeiten, deren Fokus auf der Entwicklung einer neuen Mining-Methode liegt. „Überblick“ aggregiert diejenigen Arbeiten, welche eine Gesamtsicht auf das Process Mining, oder spezielle Teilbereiche des Process Mining, in Form von State-of-the-Arts, Reviews oder

Übersichten darstellen. Unter weiterführenden Arbeiten sind Papiere zu verstehen, welche weitere Themen des Process Mining, wie beispielsweise die Evaluation der aus dem Mining resultierenden Prozessmodelle⁴⁷ bearbeiten und nicht unter die Methoden fallen. Aufgrund der sehr unterschiedlichen Themen der weiterführenden Arbeiten, werden diese zwar genannt, jedoch nicht tiefgründig analysiert. Die für die nachfolgenden Ergebnisse zugrunde liegenden Übersichts- und Analysetabellen sind dem Anhang zu entnehmen

In der Kategorie „Überblick“ wurden folgende Arbeiten identifiziert:

| Autor | Jahr | Titel |
|-----------------------------------|-------------|---|
| Van der Aalst und Weijters | 2003 | Process Mining: A Research Agenda |
| Van der Aalst et al. | 2003 | Workflow Mining: A survey of issues and approaches |
| Herbst und Karagiannis | 2004 | Workflow mining with InWoLvE |
| Tiwari und Turner | 2008 | A review of business process mining: state-of-the-art and future trends |
| Van der Aalst | 2008 | Decision Support Based on Process Mining |
| Dongen et al. | 2009 | Process Mining: Overview and Outlook of Petri Net Discovery Algorithms |

Tabelle 4: Arbeiten in der Kategorie "Überblick"

Interessant ist hierbei, dass sich alle genannten Arbeiten, abgesehen von van der Aalst 2008⁴⁸, ausschließlich mit der Prozesssicht befassen. Dessen Arbeit beschäftigt sich darüber hinaus mit der Fallsicht im Sinne der Entscheidungsunterstützung in Geschäftsprozessen. Die nachfolgenden Ausführungen machen deutlich, dass dieser Sachverhalt auf die sehr geringe Anzahl der Arbeiten in den anderen Bereichen (Organisations- und Fallsicht) zurückzuführen ist. Die Arbeit von Dongen et al.⁴⁹ ist darüber hinaus die einzige, die sich ausschließlich mit Methoden befasst, deren Output ein Petri-Netz ist.

⁴⁷ Siehe: Rozinat, A., Veloso, M., van der Aalst, W.M.P.: Using Hidden Markov Models to Evaluate the Quality of Discovered Process Models, In: Proceedings of Induction of Process Models IPM workshop at ECML PKDD 2008.

⁴⁸ van der Aalst: Decision Support Based on Process Mining, In: International Handbooks on Information Systems, 2008, IV, 637-657.

⁴⁹ van Dongen, B.F., de Medeiros, A.K., Wen, L.: Process Mining: Overview and Outlook of Petri Net Discovery Algorithms, In: Lecture Notes in Computer Science, 2009, Volume 5460/2009.

Für die Identifikation der Papiere im Bereich der „Methoden“ wurde insbesondere die Arbeit von Tiwari und Turner⁵⁰ herangezogen, da diese ebenfalls eine Literaturübersicht darbietet. Obwohl die Arbeit 2008 erschienen ist, wurden lediglich Arbeiten im Zeitraum zwischen 1998 und 2005 betrachtet, welche hier übernommen und auf die gestellten Fragestellungen abgebildet wurden. Dabei wurden die Angaben der Arbeit überprüft und gefundene Unstimmigkeiten (beispielsweise fehlerhafter Zuordnung der Output-Modelle oder Mehrfachnennungen) korrigiert. Im Zuge der detaillierten Betrachtung dieser Arbeit, wurde festgestellt, dass die dargebotene Übersicht nicht vollständig ist, wodurch auch im genannten Zeitraum weitere Arbeiten ergänzt werden konnten. Die früheste identifizierte Arbeit stammt dabei aus dem Jahr 1995 (Cook und Wolf⁵¹), was somit als „Beginn“ des Process Mining interpretiert werden kann.

Abbildung 3 visualisiert den zeitlichen Verlauf der identifizierten Arbeiten zwischen 1995 und 2010. Die Jahre 2004 und 2005 stehen aufgrund der verhältnismäßig großen Anzahl besonders heraus.

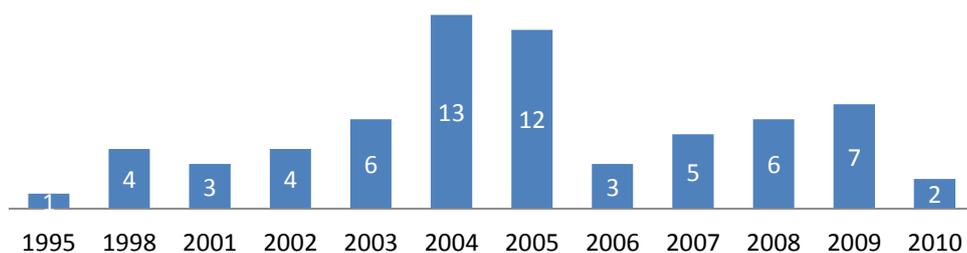


Abb. 3: Zeitlicher Verlauf der „Methoden“-Entwicklung

Alle Arbeiten wurden zunächst auf die behandelte Sichtweise und die zugrunde liegende Mining-Methode untersucht, anschließend wurde der resultierende Modelltyp festgehalten.

Durch diese Analyse konnte die bereits angesprochene besondere Bedeutung der Prozesssicht im Process Mining belegt werden. Während sich 88% der identifizierten Arbeiten mit dieser Sicht beschäftigten, behandelten lediglich 9% die Organisations- und 3% die Fallsicht (siehe Abb. 4).

⁵⁰ Tiwari, A., Turner, C.J.: A review of Business Process Mining – State-of-the-Art and Future Trends, In: Business Process Management Journal, Vol. 14.

⁵¹ Cook, J.E., Wolf, A.L.: Automating Process Discovery through Event-Data Analysis, In: ICSE '95 Proceedings of the 17th international conference on Software engineering.

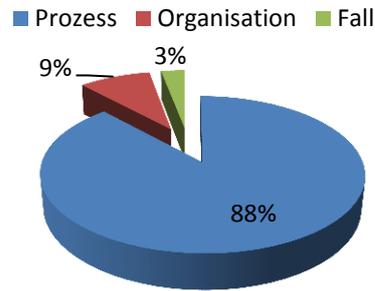


Abb. 4: Behandelte Sichten im Process Mining

Ebenfalls konnte gezeigt werden, dass der Ursprungsgedanke des Process Mining in der Prozesssicht zu finden ist, während die Anfänge der Bearbeitung der Organisationssicht auf 2004 und die der Fallsicht auf 2006 datiert werden können. In Abb. 5 ist weiter auffällig, dass im Jahr 2009 scheinbar 9 Arbeiten gezählt werden, während in Abb. 3 lediglich 7 Arbeiten erscheinen. Dies hängt mit der Tatsache zusammen, dass sich manche Arbeiten mit verschiedenen Sichtweisen, ebenso mit unterschiedlichen Mining-Methoden und Output-Modellen befassen. Um diese Fälle zu berücksichtigen, wurden die entsprechenden Arbeiten mehrfach aufgeführt.

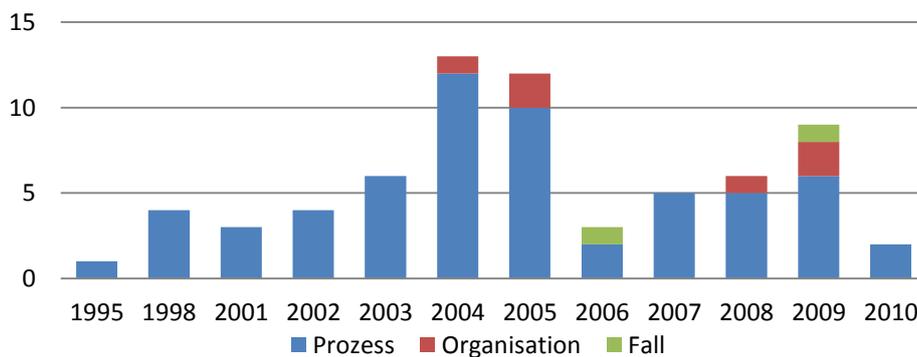


Abb. 5: Zeitlicher Verlauf der Sichten-Bearbeitung

Im Bereich der verwendeten Mining-Methoden hat sich gezeigt, dass mit 66% die Mehrzahl der Arbeiten rein algorithmische Ansätze, beispielsweise in Form von Ordnungsrelationen oder Metriken, verfolgen. Weiterhin sind mit 29% die hybriden Ansätze sehr oft vertreten, was mitunter auf die Potentiale der Rauschdatenbehandlung zurückzuführen ist. Rein statistische Ansätze sind hingegen mit 2% sehr selten zu finden, ebenso die semantischen Ansätze mit 3%.

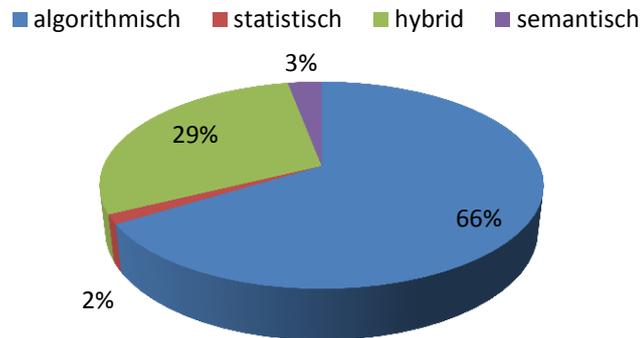


Abb. 6: Verwendete Mining-Methode

Wie bereits in Kapitel 3.7 angesprochen, konnte nun belegt werden, dass es sich bei den semantischen Ansätzen um ein sehr neues Themengebiet im Process Mining handelt, was eine Begründung für die bisher seltene Verwendung darstellt. Erste Arbeiten dazu konnten im Jahr 2008 gesichtet werden, wobei es sich außerhalb des Bereichs „Methoden“ im Wesentlichen um erste Ansätze zur Formalisierung handelt. Abbildung 7 kann weiterhin entnommen werden, dass rein statistische Ansätze heute kaum eine Rolle spielen.

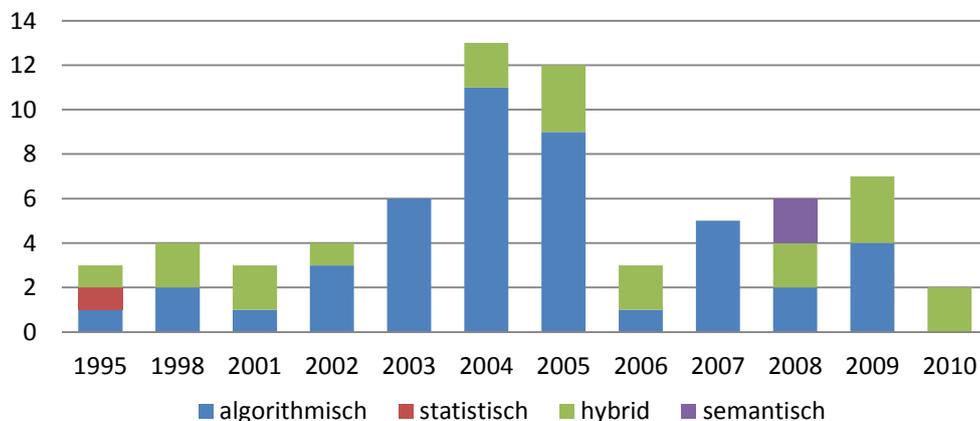


Abb. 7: Zeitlicher Verlauf der verwendeten Mining-Methoden

Im Bereich der Output-Modelle konnte zwischen EKP (= Ereignisgesteuerte Prozesskette), Petri-Netz, FSM (= Finite State Maschine – dt. endlicher Zustandsautomat), CFG (Control-Flow Graph), „Block-oriented Model“ und Soziogrammen unterschieden werden. Weitere Typen wurden als „Sonstige“ deklariert, worunter beispielsweise Bearbeiterzuordnungsregeln im Falle von Ly gehören. Teilweise beschäftigten sich die Autoren auch mit bestimmten Prob-

lemstellungen im Process Mining, wodurch zwar Lösungsansätze erarbeitet wurden, jedoch keine konkrete Modellierung der Ergebnisse erfolgte. In diesen Fällen wurde die Bezeichnung „unbekannt“ gewählt.

Wie bereits erwähnt, existiert bis heute kein Standard, welcher den Output bzw. die Visualisierung der Mining-Ergebnisse definiert. Dadurch kommen unterschiedliche Repräsentationsformen zum Einsatz, was nicht immer sinnvoll erscheint. Beispielsweise sind die Modelltypen untereinander in vielen Fällen nicht transformierbar, wodurch eine Methoden-Evaluation erheblich erschwert wird.

Abbildung 8 zeigt jedoch, dass Petri-Netze in 46% der Fälle verwendet werden, CFGs in 21% und EPKs in 6% der Fälle. Hierbei muss beachtet werden, dass diese Typen, unter bestimmten Bedingungen, relativ einfach untereinander transformierbar sind und deshalb als kompatibel angesehen werden können. Diese drei Modelltypen werden demnach in 73% der identifizierten Methoden verwendet.

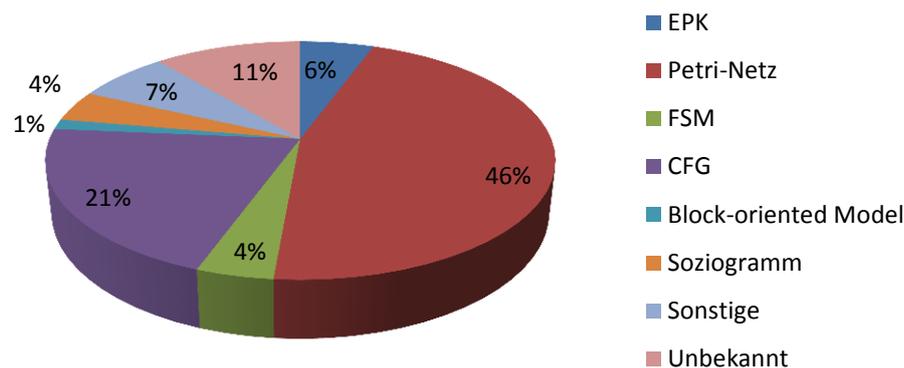


Abb. 8: Verwendete Modelltypen

Mit 1% und 4% können blockorientierte Modelle und Zustandsautomaten (FSM) als Exoten angesehen werden, während es sich bei den Soziogrammen um Sonderfälle handelt. Soziogramme bilden im Process-Mining Kommunikations- und Organisationsstrukturen ab, sodass sie in den Bereich des Organizational Mining fallen. Da das organisatorische Mining jedoch erst seit 2004 näher betrachtet wird, kann hier nicht von einem Exoten gesprochen werden, sondern vielmehr von einem gängigen Modelltyp in diesem Bereich.

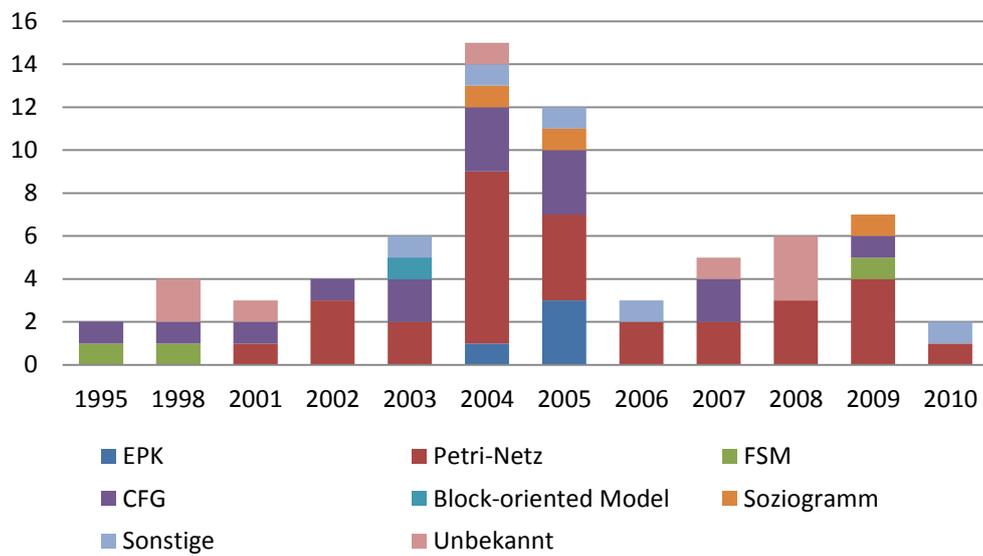


Abb. 9: Zeitlicher Verlauf der verwendeten Modelltypen

4 State-of-the-Art: Process Mining und Organisationswissen

4.1 Einführung und Überblick

Wie in Kapitel 3.9 gezeigt wurde, konzentrieren sich die bisherigen Arbeiten zum Process Mining hauptsächlich auf die Prozesssicht. Als mögliche Begründung dafür nennt Ly⁵² die sehr begrenzten Potentiale, um aus Verlaufsdaten auf organisatorische Aspekte zu schließen. Er bezeichnet dies als eine unrealistische Aufgabe, da die Verlaufsdaten typischerweise nicht genügend Informationen für diesen Zweck enthalten. Weiterhin sei es fraglich, ob Organisationswissen aus der Sicht eines Prozesses konstruiert werden sollte.

Aus den genannten Gründen beschäftigen sich die vorhandenen Arbeiten zum Organizational Mining auch weniger damit, Organisationsmodelle aus den Verlaufsdaten abzuleiten, sondern vielmehr mit den Beziehungen zwischen den Akteuren und den Beziehungen zwischen Aufbauorganisation und Prozess.

Zu der ersten Kategorie, den Beziehungen zwischen den Akteuren, sind die Mehrzahl der relevanten Arbeiten zu zählen. Es wird versucht, ein soziales Netzwerk in Form von Soziogrammen abzuleiten, welches wiederum als Eingabe für Netzwerkanalysen (hier: „Social Network Analysis“ = SNA) verwendet werden kann. Diesen Weg verfolgen sowohl van der Aalst und Song⁵³ als auch Ang et al.⁵⁴ und van der Aalst et al.⁵⁵. Im Zuge der Ableitung von Soziogrammen aus den Verlaufsdaten beschäftigten sich die Autoren mit der Erarbeitung unterschiedlicher Metriken. Bei van der Aalst und Song handelt es sich beispielsweise konkret um Metriken, basierend auf Kausalität, Prozessinstanzen, gemeinsamen Aktivitäten und besonderen Ereignistypen (mehr dazu in Kapitel 4.2.1).

⁵² Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 57, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

⁵³ van der Aalst, W.M.P., Song, M. (2004), Mining social networks: uncovering interaction patterns in business processes, in Desel, J., Pernici, B. and Weske, M. (Eds), International Conference on Business Process Management (BPM 2004), Springer Verlag, Berlin.

⁵⁴ Ang Gao, Yang Yang, Ming Zeng, Jing-Le Zhang, Yue-Wei Wang. Organizational Structure Mining Based on Workflow Logs, In: International Conference on Business Intelligence and Financial Engineering 2009 (BIFE '09).

⁵⁵ Wil M.P. van der Aalst, Hajo A. Reijers, Minseok Song: Discovering Social Networks from Event Logs, In: Computer Supported Cooperative Work (CSCW) Volume 14, Number 6, 549-593.

Die formale Netzwerkanalyse unterscheidet zwischen drei wesentlichen Aspekten⁵⁶:

1. Analysen zur Netzwerkstruktur, z.B. Netzdichte und Gliederung in Teilnetze
2. Analyse von Knoten, z.B. Zentralität eines Individuums
3. Analysen zur Art und den Eigenschaften der Beziehungen, z.B. Symmetrie und Transitivität

Diese Analysen können auf Basis der resultierenden Soziogramme mit gängigen Anwendungen, wie beispielsweise NetMiner, iGraph oder InFlow durchgeführt werden. Für eine Einführung in die Netzwerkanalyse soll an dieser Stelle auf die Arbeit von Diaz-Bone⁵⁷ verwiesen werden. Eine Übersicht der vorhandenen Softwarelösungen ist bei Wikipedia⁵⁸ zu finden (siehe Kapitel 4.2.2).

In der zweiten Kategorie, welche sich mit den Beziehungen zwischen Aufbauorganisation und Prozess befasst, konnte bisher nur die Arbeit „Mining Staff Assignment Rules from Event-Based Data“⁵⁹ von Ly et al. identifiziert werden, welche in den direkten Kontext des Process Mining einzuordnen ist. Auf Basis von Verlaufsdaten und eines a-priori vorhandenen Organisationsmodells wird versucht, Bearbeiterzuordnungsregeln zu konstruieren. Dazu wird zunächst das Organisationsmodell mit den Verlaufsdaten verknüpft (Voraussetzung hierfür ist das Vorhandensein der Initiator-Information in den Verlaufsdaten), anschließend werden auf Basis unterschiedlicher Metriken die notwendigen Qualifikationen für die Ausführung der Aktivitäten ermittelt, wodurch die Generierung der Regeln ermöglicht wird (mehr dazu in Kapitel 4.3.4).

Abbildung 10 veranschaulicht die beiden vorgestellten Varianten.

Grundsätzlich kann jedoch im Bereich der Beziehungen zwischen Aufbauorganisation und Prozess eine Vielzahl weiterer Literatur identifiziert werden, welche sich allerdings nicht im direkten Process Mining-Umfeld bewegt. Im Wesentlichen geht es dabei um die Kopplung von Aufbau- und Ablauforganisation und darum, wie diese Kopplung sinnvoll in Workflow-Management Systemen umgesetzt und eingesetzt werden kann.

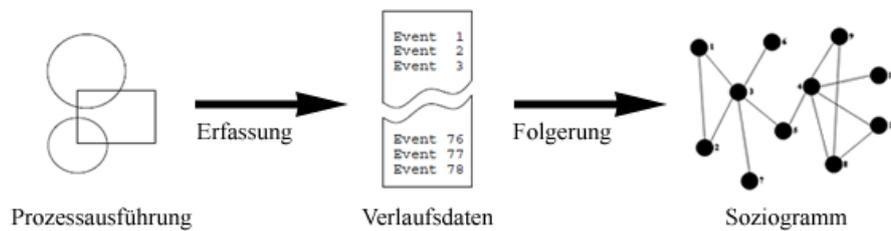
⁵⁶ Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 58, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

⁵⁷ Diaz-Bone, R.: Eine kurze Einführung in die sozialwissenschaftliche Netzwerkanalyse, In: Mitteilungen aus dem Schwerpunktbereich Methodenlehre, Heft Nr. 57, Berlin 2006, URL: http://www.rainer-diaz-bone.de/Diaz-Bone_Netzwerkanalyse.pdf, Abrufdatum: 09.06.2011.

⁵⁸ Social Network Analysis – Wikipedia, URL: http://en.wikipedia.org/wiki/Social_network_analysis_software, Abrufdatum: 09.06.2011.

⁵⁹ Ly, T.L., Rinderle, S., Dadam, P., Reichert, M.: Mining Staff Assignment Rules from Event-Based Data, In: 3rd International Conference on Business Process Management 2005.

Beziehungen zwischen Akteuren: *van der Aalst und Song, Ang et al., van der Aalst et al.*



Beziehungen zwischen Aufbauorganisation und Prozess: *Ly et al.*

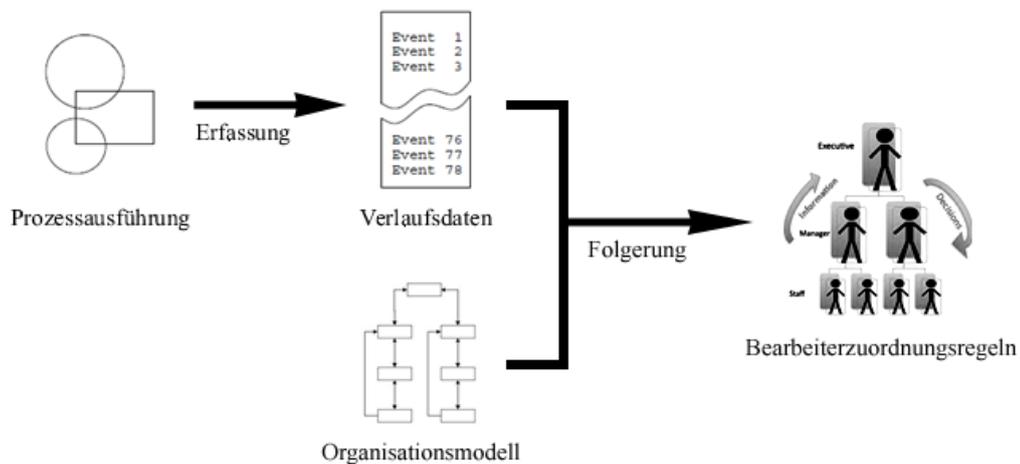


Abb. 10: Vorgehen im Organizational Mining

Beispielsweise muss geklärt werden, welche organisatorischen Konstrukte für die Ausführung eines Prozesses relevant sind, welche Strukturbeziehungen umsetzbar sein sollen und welche Einsatzmöglichkeiten des Organisationswissens im Hinblick auf die Prozessausführung denkbar sind.

Nachfolgend wird zunächst das Mining der Beziehungen zwischen den Akteuren an der Arbeit von van der Aalst und Song exemplarisch erläutert. Anschließend werden die Beziehungen zwischen Aufbau- und Ablauforganisation im Detail diskutiert, um dann auf die Ableitung von Prozesswissen unter der Verwendung von Verlaufsdaten und Organisationswissen einzugehen.

4.2 Beziehungen zwischen Akteuren

4.2.1 „Mining Social Networks“

Zur Erläuterung des gängigen Vorgehens im Bereich Organizational Mining wird im Folgenden die Methode von van der Aalst et al. vorgestellt. Wie einführnd beschrieben, basiert diese, wie auch die anderen bisher identifizierten

Methoden, auf verschiedenen Metriken, sodass die Vorstellung der Arbeit „Mining Social Networks“⁶⁰ ausreichend für die Verdeutlichung des Vorgehens in dieser Themenfacette ist.

Das Ziel der Arbeit besteht in der Ableitung eines Soziogramms aus den Verlaufsdaten der an dem Prozess beteiligten Informationssysteme. Dieses Soziogramm soll anschließend als Eingabe für gängige Software im Bereich der SNA dienen.

Um sich dem Problem zu nähern, haben die Autoren vier Typen von Metriken identifiziert – basierend auf möglicher Kausalität, gemeinsamen Prozessinstanzen, gemeinsamen Aktivitäten und besonderen Ereignistypen.

Metriken basierend auf (möglicher) Kausalität betrachten den Verlauf von Prozessinstanzen im Hinblick auf die beteiligten Akteure. Dazu wurden die beiden Metriken „handover of work“ und „in-between“ definiert. Ein „handover of work“ von Akteur A zu Akteur B existiert in einer Prozessinstanz dann, wenn bei zwei aufeinanderfolgenden Aktivitäten die erste von A und die zweite von B ausgeführt wurde. Neben der Betrachtung der direkten Nachfolger können auch indirekte Nachfolger mit Hilfe des „causality fall factor“ β betrachtet werden. Existieren beispielsweise 3 Aktivitäten zwischen einer Aktivität, die von A ausgeführt wurde und einer Aktivität, die von B ausgeführt wurde, wäre der „causality fall factor“ β^3 . Eine entsprechende Metrik wäre das „subcontracting“, bei welcher gezählt wird, wie oft B eine Aktivität ausführt, die zwischen zwei Aktivitäten liegt, die von A ausgeführt werden. Dies könnte dann auf eine untergeordnete Aufgabe oder eine Teilaufgabe (engl. "subcontracting") hinweisen. Die Autoren merken an, dass diese Metriken, wie auch alle nachfolgenden, beliebig verfeinert werden können.

Metriken basierend auf gemeinsamen Prozessinstanzen zählen, wie oft zwei Akteure an Aktivitäten derselben Prozessinstanz beteiligt sind („working together metrics“). Dabei liegt die Annahme zugrunde, dass Akteure, die zusammen an mehreren Prozessinstanzen arbeiten eine stärkere Beziehungen zueinander haben, als diejenigen, die weniger oft zusammenarbeiten.

Metriken basierend auf gemeinsamen Aktivitäten betrachten die Aktivitäten, die von den Akteuren ausgeführt werden. Die Annahme liegt darin, dass Akteure, die dasselbe tun, eine stärkere Beziehung zueinander haben, als diejenigen, die vollkommen unterschiedlichen Aufgaben nachgehen. Jeder Akteur hat demnach ein Profil, welches auf der Häufigkeit der Ausführung von bestimm-

⁶⁰ van der Aalst, W.M.P., Song, M. (2004), Mining social networks: uncovering interaction patterns in business processes, In Desel, J., Pernici, B. and Weske, M. (Eds), International Conference on Business Process Management (BPM 2004), Springer Verlag, Berlin, S. 252ff.

ten Aktivitäten basiert. Der Abstand dieser Profile kann durch unterschiedliche Arten gemessen werden (beispielsweise mit Methoden der Clusteranalyse), sodass auch hier eine Vielzahl von Metriken vorstellbar ist.

Kritisch ist jedoch anzumerken, dass unterschiedliche Aktivitäten durchaus ähnlich sein können. Ly⁶¹ merkt dazu an, dass die Ähnlichkeit von Aktivitäten unter Umständen schwer zu bestimmen sei und dieser Aspekt in der vorgestellten Metrik nicht berücksichtigt wird.

Metriken basierend auf besonderen Ereignistypen betrachten dem Namen nach Ereignistypen. Dabei wird angenommen, dass sich Ereignisse auf Aktivitäten beziehen, jedoch auch andere Ereignisse, wie beispielsweise Neuzuweisungen einer Aktivität von einem Akteur zu einem anderen existieren („reassign“). Wenn beispielsweise Akteur A Aufgaben an Akteur B delegiert, der umgekehrte Fall jedoch nicht eintritt, ist es wahrscheinlich, dass A in einer hierarchisch höheren Position arbeitet als B. Aus Sicht der SNA sind solche Beobachtungen sehr interessant, da sie konkrete „Macht“-Beziehungen repräsentieren.

Bezogen auf diese Metrik muss allerdings betont werden, dass die Autoren weitere Anforderungen an die Verlaufsdaten erheben. Um Ereignistypen auswerten zu können, ist es notwendig, dass auch Ereignisse mitprotokolliert werden, was in den grundlegenden Anforderungen (siehe Kapitel 3.4) nicht verlangt wird. Da es sich bei den fokussierten Ereignistypen aktuell ausschließlich um Sonderfälle und Ausnahmen handelt, werden auch andere mögliche Variationen dieser Metrik die genannten Information benötigen.

Ein Problem des „handover of work“ besteht außerdem darin, dass ohne weitere Informationen kein aussagekräftiges Soziogramm generiert werden kann, da ggf. falsche Informationen abgeleitet werden. Der Ansatz interpretiert die Verlaufsdaten so, dass eine sequentielle Ausführung der Aktivitäten innerhalb des Prozesses angenommen wird, was häufig nicht der Realität entspricht. Betrachtet man das Beispiellog aus Tabelle 5 ergäbe sich das Soziogramm in Abbildung 11.

| Aktivität | Initiator |
|------------------|------------------|
| A | M1 |
| B | M2 |
| C | M4 |
| D | M3 |
| E | M5 |
| F | M1 |

Tabelle 5: "Mining Social Networks" - Beispiellog

⁶¹ Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 61, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

Durch die Annahme der sequenziellen Ausführung kann das „handover of work“ immer von genau einem Akteur zu genau einem anderen Akteur abgeleitet werden. Häufig existieren jedoch in einem Prozess parallel ablaufende Teilsequenzen. Da die Metriken diese Parallelität nicht erkennen können, besteht die Gefahr, dass falsche Kausalitäten abgeleitet werden.

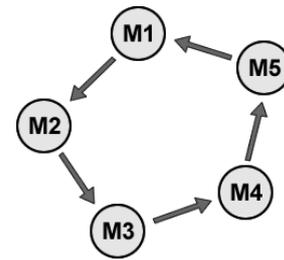


Abb. 11: Soziogramm - "handover of work"

Nun soll angenommen werden, dass die EPK in Abbildung 12 das korrekte Prozessmodell für den durch die Verlaufsdaten repräsentieren Prozess darstellt. Die Aktivitätssequenzen BD und CE laufen darin parallel ab, sodass das reale „handover of word“ von M2 zu M3 und von M4 zu M5 stattfindet.

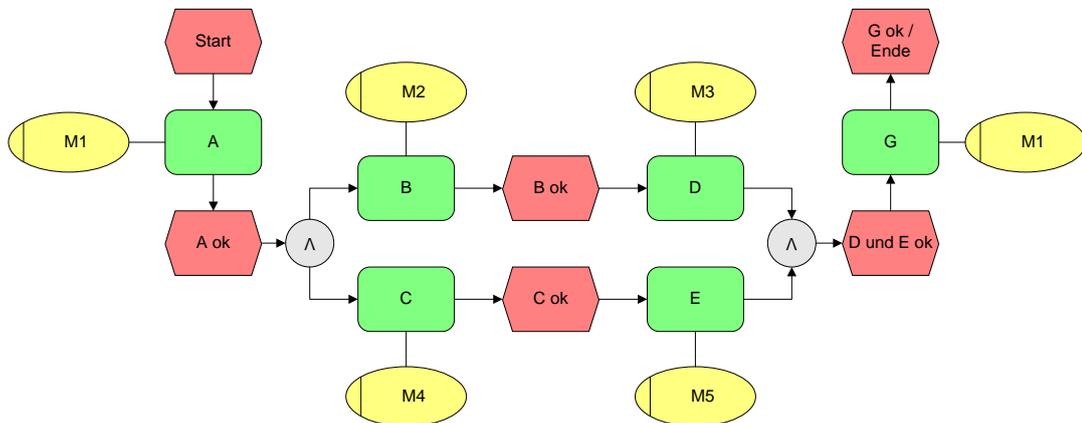


Abb. 12: EPK - Beispielprozess "Minig Social Networks"

Abbildung 13 repräsentiert das Soziogramm, welches unter Verwendung von Prozesswissen entstehen würde. Man kann leicht erkennen, dass es sich signifikant vom ursprünglichen Soziogramm unterscheidet und die ursprünglichen Fehler behebt.

Es konnte gezeigt werden, dass die vorgestellten Metriken nicht ausreichen, um das angestrebte Ziel zuverlässig zu erreichen. Dieses Problem kann jedoch durch die Vorlagerung eines Control-Flow Mining (Process Mining der Prozesssicht) angegangen werden, wobei das resultierende Prozessmodell für die Erstellung des Soziogramms genutzt wird.

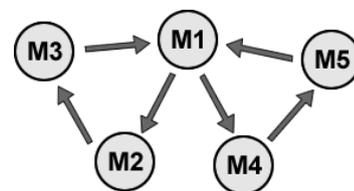


Abb. 13: Soziogramm - "handover of work" mit Prozesswissen

Es sollte allerdings beachtet werden, dass auch das aus dem vorgelagerten Process Mining resultierende Prozessmodell die Parallelität nicht in jedem Fall erkennen kann – dies ist stark von der eingesetzten Mining Methode und den Verlaufsdaten abhängig.

Das resultierende Soziogramm, welches mittels der Metriken mit gewichteten Kanten erzeugt wird, kann für weitere Analysen mit gängigen Softwarelösungen zur Netzwerkanalyse weiterverarbeitet werden.

4.2.2 Aktuelle Software

Der vorgestellte Ansatz wurde mit dem Werkzeug MiSoN⁶² (= „Mining Social Networks“) prototypisch implementiert. Es leitet die Beziehungen zwischen Akteuren aus den Log-Daten unterschiedlicher Systeme (z.B. ERP-, oder CRM-Systeme) ab und konstruiert daraus Soziogramme. Es existiert anschließend die Möglichkeit des Exports in einem Matrix-Format, welches von den meisten SNA-Werkzeugen unterstützt wird. Die Ergebnisse können so als Input für entsprechende Werkzeuge herangezogen werden.

MiSoN wurde in Java entwickelt und verwendet verschiedene XML-Bibliotheken - weiterhin steht eine grafische Benutzeroberfläche zur Verfügung (siehe Abbildung 14). Das Mining selbst geht von einem Tool-unabhängigen XML-Format aus, welches Informationen über Prozesse, Prozessinstanzen, Aktivitäten, Zeiten und Akteure zur Verfügung stellt. Für das Mining können unterschiedliche Metriken gewählt und konfiguriert werden. Beispielsweise kann ausgewählt werden, ob die Ähnlichkeit von Aktivitäten (siehe Kapitel 4.2.1) über die euklidische Distanz, Korrelationskoeffizient, Gleichheitskoeffizient oder Hamming Distance ermittelt werden soll. Für jede einzelne Metrik lässt sich die entsprechende Matrix und der passende Graph generieren. Weiterhin lässt sich auch die Prozessstruktur importieren, um den genannten Problemstellungen entgegenzuwirken.

Der Export der Ergebnisse (für SNA-Werkzeuge) beinhaltet Informationen über Anzahl und Namen der Akteure, sowie eine Beziehungsmatrix („relationship matrix“).

MiSoN wurde bis Ende Mai 2009 in der aktuellen Version entwickelt – weitere Werkzeuge in diesem Bereich konnten nicht identifiziert werden.

⁶² van der Aalst, W.M.P., Song, M. (2004), Mining social networks: uncovering interaction patterns in business processes, In Desel, J., Pernici, B. and Weske, M. (Eds), International Conference on Business Process Management (BPM 2004), Springer Verlag, Berlin, S. 256ff.

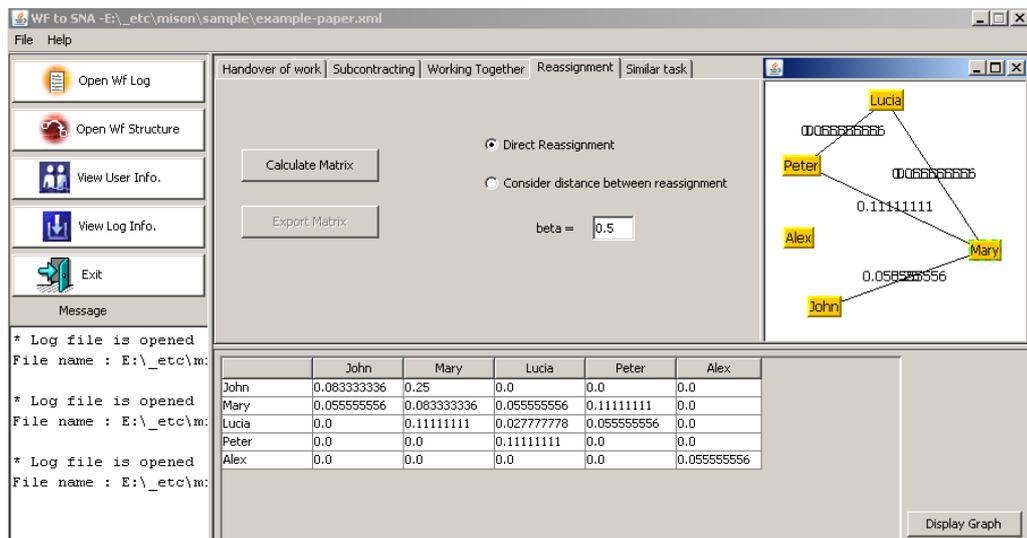


Abb. 14: Screenshot – MiSoN

4.3 Beziehungen zwischen Aufbau- und Ablauforganisation

4.3.1 Grundlagen

Um die Problem- und Fragestellungen in diesem Bereich korrekt erfassen zu können, ist es zunächst notwendig, die Terminologie zu klären und eine Abgrenzung von Aufbau- und Ablauforganisation durchzuführen.

Unter Aufbauorganisation sollen im Folgenden die strukturellen Gegebenheiten als statische Komponente einer Organisation verstanden werden.⁶³ Die Ablauforganisation bezeichnet hingegen die in einer Organisation stattfindenden Informations- und Kommunikationsprozesse als dynamische Komponente.⁶⁴ Aufgrund gegenseitiger Wechselwirkungen kann keine strikte Trennung zwischen den beiden Komponenten erfolgen, sodass die Aufbaustruktur mit den betrieblichen Funktionen und Prozessen und den relevanten Daten über die Ablaufstruktur verbunden wird.

Die Aufbauorganisation kann mit Organisationsmodellen (z.B. Organigramm) beschrieben werden, Abläufe bzw. Prozesse werden typischerweise mit Methoden wie EPK oder Petri-Netzen modelliert. Dazu ist weiter anzumerken, dass die eEPK (= erweiterte ereignisgesteuerte Prozesskette) in diesem Zu-

⁶³ Hars, A., Zimmermann, V., Scheer, A.-W.: Entwicklungsrichtlinien für die computergestützte Modellierung von Aufbau- und Ablauforganisation, IWi-Heft 105, Dezember 1993, S. 9.

⁶⁴ Wedekind, E.E.: Informationsmanagement in der Organisationsplanung, Deutscher Universitätsverlag, S. 19.

sammenhang als besonders aussagekräftig zu bewerten ist, da organisatorische Elemente in der Methodendefinition vorhanden sind und dadurch eine Kopp- lung der Organisationsicht bereits vorgesehen ist.

Da Organisationsmodelle organisatorische Einheiten und deren Strukturen be- schreiben, handelt es sich ebenfalls um Komponenten eines betrieblichen In- formationssystems, wodurch sie sich in das ARIS⁶⁵-Haus (=Architektur Inte- griertier Informationssysteme) einordnen lassen.⁶⁶ Sie werden auf Fachkonzept- ebene in der Organisationsicht beschrieben, während die fachliche Beschrei- bung der Ablauforganisation in der Steuerungssicht erfolgt, welche eine Ver- knüpfung von Daten, Funktionen und Organisationseinheiten darstellt. Siehe dazu Abbildung 15.



Abb. 15: Aufbau- und Ablauforganisation im ARIS-Haus⁶⁷

4.3.2 Potentiale des Organisationswissens

Für die sinnvolle Verwendung von Organisationswissen im Process Mining ist es hilfreich, die Potentiale und Anwendungsmöglichkeiten des Organisations- wissens zu kennen. Die Idee besteht darin, dass aus bereits vorhandenen An-

⁶⁵ Scheer, A.-W.: Architektur integrierter Informationssysteme. Springer, Berlin 1992.

⁶⁶ Hars, A., Zimmermann, V., Scheer, A.-W.: Entwicklungsrichtlinien für die computergestütz- te Modellierung von Aufbau- und Ablauforganisation, IWi-Heft 105, Dezember 1993, S. 11.

⁶⁷ In Anlehnung an: Scheer, A.-W.: Architektur integrierter Informationssysteme. Springer, Berlin 1992, S. 18.

wendungen Regeln bzw. Metriken für das Process Mining, basierend auf diesem Wissen, abgeleitet werden können. Hars et al.⁶⁸ geben dazu einen umfangreichen Überblick, welcher nun vorgestellt werden soll.

- *Organisationsanalyse*: Im Hinblick auf die Gestaltung und Kontrolle sollen Aussagen über die Organisation abgeleitet werden. Die Fragestellungen reichen dabei von der Ermittlung einfacher Kenngrößen (z.B. Leitungsspanne) bis hin zu methodenintensiven Auswertungen, welche Schwachstellen und Fehlentwicklungen in Aufbau- und Ablauforganisation feststellen können. Weiterhin sind Vergleiche mit Organisation-Referenzmodellen möglich.
- *Organisationsplanung und -gestaltung*: Ziel ist die Verbesserung bestehender Organisationsstrukturen auf Basis der vorhandenen Organisationsmodelle. Beispielsweise könnten alternative Szenarien und Sollkonzepte mit Simulationsverfahren auf ihre mögliche Effektivität geprüft werden.
- *Workflow-Unterstützung*: Während der Prozessausführung können potentielle Bearbeiter und Ressourcen identifiziert und zugewiesen werden. Hierdurch ergibt sich ein enger Zusammenhang zwischen Organisations- und Prozessmodell, da sich jede Änderung an der Struktur (und dem entsprechenden Modell) am jeweils anderen Modell niederschlägt.
- *Ableitung von Anforderungen an ein betriebliches IuK* (= Informations- und Kommunikationssystem): Das Organisationsmodell enthält Kommunikationsbeziehungen und beschreibt Prozesse, die aus einzelnen Tätigkeiten bestehen. Ziel ist demnach die Ableitung des DV-Konzeptes eines Informationssystems aus den Aufbau- und Ablauforganisationsmodellen.
- *Auswahl und Konfiguration von EDV-Systemen*: Hierbei handelt es sich um eine Erweiterung der zuvor genannten Anwendung. Während diese für die Entwicklung eines neuen IS (=Informationssystem) genutzt wird, soll hier ein bereits vorhandenes ausgewählt und konfiguriert werden. Dazu muss beispielsweise das Organisationsmodell mit dem Modell der Software verglichen werden.
- *Ausbildung und Einarbeitung von Mitarbeitern*: Anhand von Organisationsmodellen kann den Mitarbeitern ein ganzheitlicher Überblick über das Unternehmen gegeben werden. Zusammen mit Prozessmodellen

⁶⁸ Hars, A., Zimmermann, V., Scheer, A.-W.: Entwicklungsrichtlinien für die computergestützte Modellierung von Aufbau- und Ablauforganisation, IWi-Heft 105, Dezember 1993, S. 16ff.

können sie das Vorgehen und die Verantwortlichkeiten kennen lernen. Das verdeutlicht auch die eigene Rolle in der Vorgangskette.

- *Entwicklung neuer Organisationskonzepte*: Organisationsmodelle können durch hohe Transparenz und Verfügbarkeit Auswirkungen auf Gestaltungskonzepte haben. So können beispielsweise sehr einfach interdisziplinäre Teams aufgestellt oder zusammengehörige Tätigkeiten gruppiert werden.
- *Support-Unterstützung*: Durch die Weitergabe von Organisationsmodellen an Kunden, können diese im Supportfall einen passenden Mitarbeiter identifizieren, wodurch Ressourcen geschont und die Abwicklung beschleunigt wird. Dies kann auch auf die unternehmensübergreifende Zusammenarbeit übertragen werden.
- *Mitarbeiterentwicklung*: Organisationen beinhalten im Allgemeinen Informationen über die Qualifikationen von Mitarbeitern. Diese liefern entscheidungsunterstützende Informationen für die Auswahl von Mitarbeitern im Hinblick auf notwendige Entwicklungsmaßnahmen.

Einige dieser Anwendungsmöglichkeiten liefern unmittelbare Ansätze für Überlegungen zum Process Mining. Beispielsweise weist die *Workflow-Unterstützung* auf die Grundidee des Ansatzes zum „Staff Assignment Mining“ von Ly hin (siehe Kapitel 4.3.4). Ebenfalls trägt dazu der Aspekt *Mitarbeiterentwicklung* bei, da hier auf die Qualifikationen von Mitarbeitern eingegangen wird. Eine Idee könnte darin bestehen, dass sich die Aktivitäten innerhalb eines Prozesses über die dafür notwendigen Qualifikationen definieren lassen.

4.3.3 Organisations-Metamodelle

Bis heute existiert kein allgemein anerkannter Standard für die Modellierung der Aufbauorganisation. Häufig werden die Konstrukte, bzw. das Organisations-Metamodell im Hinblick auf den zu erfüllenden Zweck abgestimmt und somit den Anforderungen angepasst. Daraus resultiert eine Vielzahl von Organisations-Metamodellen, welche mehr oder weniger brauchbar für die Erreichung unterschiedlicher Zielsetzungen erscheinen.

Die bekanntesten dieser Organisations-Metamodelle sollen im Folgenden einander gegenübergestellt und mit Blick auf das Process Mining beurteilt werden.

Zur Gegenüberstellung werden die Metamodelle, in Anlehnung an Rosemann und zur Mühlen,⁶⁹ zunächst auf die Existenz unterschiedlicher (workflow-relevanter) Konstrukte untersucht, welche im Folgenden kurz vorgestellt werden.

Eines der wichtigsten Konstrukte stellt die Rolle dar, welche einerseits die für die Ausführung einer Aktivität notwendige Qualifikation beschreibt und andererseits die an den Rolleninhaber übertragenen Kompetenzen darstellt. Eine Stelle bezeichnet die Zusammenfassung von Rollen (Aufgaben), welche insgesamt die Arbeitszeit einer Person (eines Mitarbeiters) auslasten. Dabei handelt es sich um das klassische Konstrukt zur Kopplung von Aufbau- und Ablauforganisation. Stellen werden zu Organisationseinheiten (= OE), wie beispielsweise Vertrieb oder Einkauf zusammengefasst. Organisationseinheiten können dauerhaft oder temporär (Projekte) angelegt sein.

Demzufolge soll die Existenz folgender Konstrukte überprüft werden:

- Rolle: Qualifikation, Kompetenz, Rollenhierarchie
- Person
- Organisationseinheit: dauerhaft, temporär, OE-Hierarchie
- Stelle: Stellenhierarchie, Stellentyp

Es wurde in unterschiedlichen themenspezifischen Arbeiten nach relevanten Quellenangaben gesucht, was zu nachfolgenden Ergebnissen führte: Einige der Arbeiten, z.B. Rosemann und zur Mühlen, konzentrieren sich ausschließlich auf Organisations-Metamodelle im Allgemeinen, während andere Arbeiten, z.B. Galler, ihren Fokus auf Organisationsmodelle im Workflow-Management legen. Ly hingegen befasst sich mit der Thematik im Hinblick auf eine neue Methode im Themenfeld des Process Mining, weshalb darin diverse Parallelen zur vorliegenden Arbeit bestehen.

Interessant ist diese Literaturzusammenstellung gerade deshalb, weil ggf. Auswirkungen der unterschiedlichen Zielsetzungen auf das resultierende Metamodell aufgedeckt werden können.

⁶⁹ Rosemann, M.; zur Mühlen, M.: Modellierung der Aufbauorganisation in Workflow-Management-Systemen: Kritische Bestandsaufnahme und Gestaltungsvorschläge, <http://www.b2b-workflow.de/Publications/PDF/MIRO.MIZU-EMISA%281997%29.pdf>, Aburfdatum: 14.06.2010, S. 2ff.

| Autor | Erscheinungsjahr |
|---|-------------------------|
| Braun⁷⁰ | 2007 |
| Derungs et al.⁷¹ | 1996 |
| Esswein⁷² | 1992 |
| Galler⁷³ | 1995 |
| Heilmann und Munz⁷⁴ | 1997 |
| Joosten⁷⁵ | 1996 |
| Ly⁷⁶ | 2005 |
| Rosemann und zur Mühlen⁷⁷ | 1998 |
| Scheer⁷⁸ | 2001 |

Tabelle 6: Arbeiten zu Organisations-Metamodellen

Die in den Arbeiten vorgeschlagenen Organisations-Metamodelle wurden nun entsprechend der vorgestellten Konstrukte analysiert (siehe Tabelle 7). Auffällig ist hierbei, dass nahezu alle Arbeiten die Kern-Komponenten Rolle, Person, Organisationseinheit und Stelle beinhalten, während es große Unterschiede in den jeweils untergeordneten Aspekten gibt. Nur wenigen Modelle arbeiten bei-

⁷⁰ Braun, C.: Modellierung der Unternehmensarchitektur - Weiterentwicklung einer bestehenden Methode und deren Abbildung in einem Meta-Modellierungswerkzeug, Logos 2007, S. 119.

⁷¹ Derungs, M.; Vogler, P.; Österle, H.: Metamodell Workflow. Bericht Nr. 3 (IM HSG/CC PSI/3), Kompetenzzentrum Prozess- und Systemintegration, Version 1.5, St. Gallen, 1996.

⁷² Esswein, W.: Das Rollenmodell der Organisation: Die Berücksichtigung aufbauorganisatorischer Regelungen in Unternehmensmodellen, In: Augsburg, W.; Sinz, E.J.: Bamberger Beiträge zur Wirtschaftsinformatik, Nr. 14, Bamberg 1992, S. 11.

⁷³ Galler, J.: Metamodelle des Workflow-Managements, In: IWi-Heft 121, Dezember 1995.

⁷⁴ Heilmann, H.; Munz, D.: Die Integration der Aufbauorganisation in Workflow-Management-Systeme: Konzeption eines Metamodells und Entwicklung eines Prototyps, In: Software-Labor der Universität Stuttgart Bericht SL-3/97.

⁷⁵ Joosten, S.M.M.: WorkPAD: a Conceptual Framework for Process Analysis and Design, In: ACM Transactions on Office Information Systems 1996.

⁷⁶ Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 68, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

⁷⁷ Rosemann, M.; zur Mühlen, M.: Modellierung der Aufbauorganisation in Workflow-Management-Systemen: Kritische Bestandsaufnahme und Gestaltungsvorschläge, <http://www.b2b-workflow.de/Publications/PDF/MIRO.MIZU-EMISA%281997%29.pdf>, Abrufdatum: 14.06.2010, S. 3f.

⁷⁸ Scheer, A.-W.: ARIS - Modellierungsmethoden, Metamodelle, Anwendungen, Vierte Auflage, Springer 2001, S. 56.

spielsweise mit einer Rollenhierarchie, obwohl diese überaus wichtig sein kann, um Vererbungen, und damit eine Struktur darzustellen. Fehlt diese Hierarchie, ist es nicht ohne weiteres möglich verschiedene Rollen zu aggregieren oder zu erweitern. In diesem Fall müsste jede Rolle mit der Gesamtheit der umfassenden Qualifikationen und Kompetenzen erfasst werden, obwohl andere Rollen evtl. ganze Teilmengen repräsentieren würden. Dieser Aspekt kann ebenso auf die Stellenhierarchie übertragen werden, während eine Hierarchie in Organisationseinheiten in fast allen Arbeiten gegeben ist.

| | Joosten | Derungs et al. | Galler | Rosemann und zur Mühlen | Ly | Heilmann und Kunz | Scheer | Braun | Esswein |
|-----------------------------|---------|----------------|--------|-------------------------|----|-------------------|--------|-------|---------|
| Rolle | x | x | | x | x | x | x | x | x |
| - Qualifikation | x | | x | x | x | | x | x | x |
| - Kompetenz | x | | | x | | | | x | |
| - Rollenhierarchie | | | | x | x | | | | x |
| Person | x | | x | x | x | x | | x | x |
| Organisationseinheit | x | x | x | x | x | x | x | x | x |
| - dauerhaft | x | x | x | x | x | | x | x | x |
| - temporär (Projekt) | | | | x | | | | | |
| - OE-Hierarchie | x | x | x | x | x | | x | x | |
| Stelle | x | x | x | x | x | | x | x | x |
| - Stellenhierarchie | | | x | x | | | | | |
| - Stellentyp | x | | x | x | | | | x | |

Tabelle 7: Analyse - Organisations-Metamodelle⁷⁹

Weiterhin fällt die eher seltene Integration der Kompetenz auf, was mit dem engen Zusammenhang zur Qualifikation begründet werden könnte. Gewisser-

⁷⁹ In Anlehnung an: Rosemann, M.; zur Mühlen, M.: Modellierung der Aufbauorganisation in Workflow-Management-Systemen: Kritische Bestandsaufnahme und Gestaltungsvorschläge, <http://www.b2b-workflow.de/Publications/PDF/MIRO.MIZU-EMISA%281997%29.pdf>, Ab-rufdatum: 14.06.2010, S. 10, Tab. 2: Synope der Organisationsmodelle ausgewählter Referenz-Metamodelle.

maßen kann die Kompetenz aus den Qualifikationen abgeleitet oder mit dieser geklärt werden. Ein konkretes Beispiel dafür wäre die Umschreibung der Kompetenz „LKW fahren“ als Qualifikation „darf LKW fahren“. Je nach Umsetzung könnte die Kompetenz also als Verfeinerung der Qualifikation verstanden werden.

Um ein scheinbar kritisches Konstrukt handelt es sich außerdem bei temporären Organisationseinheiten, welche ausschließlich von Rosemann und zur Mühlen umgesetzt werden. Diese sind zwar ohne Zweifel existent, jedoch lässt sich über die Notwendigkeit eines eigenen Konstrukts durchaus diskutieren. Es handelt sich ebenso wie dauerhafte Organisationseinheiten um Organisationseinheiten, mit dem einzigen Unterschied, dass diese nur für einen, in der Regel fest definierten, Zeitraum existieren. Demnach könnte man diese zeitliche Einschränkung einerseits als Attribut modellieren (Kompromisslösung), andererseits auch ersatzlos streichen. Dies hätte zur Folge, dass temporäre Organisationseinheiten als „normale“ Organisationseinheiten dargestellt würden, was in vielen Anwendungsszenarien, so auch beim Process Mining, unproblematisch wäre. Da im Process Mining die Ist-Situation abgebildet werden soll, spielt es keine Rolle ob eine Organisationseinheit temporär oder von Dauer ist, relevant ist lediglich die Existenz.

Das Referenz-Metamodell von Rosemann und zur Mühlen soll aufgrund der Vollständigkeit zu den Untersuchungsaspekten vorgestellt werden, um eine mögliche Realisierung zu veranschaulichen (siehe Abbildung 16).

Zusätzlich zu den genannten Konstrukten ist in diesem Modell auf die Beziehung „Stellvertreter“ hinzuweisen, welche zusätzlich die Modellierung der Struktur zwischen Personen ermöglicht. Die Autoren weisen hierbei allerdings auf eine Problematik im Sinne der Umsetzung hin. Viele Workflow-Management-Systeme interpretieren eine Stellvertretung als die Vererbung sämtlicher Rollen des Stelleninhabers an den Stellvertreter⁸⁰. Das ist allerdings nicht immer sinnvoll, da es auch denkbar ist, dass ein Stellvertreter nur einen bestimmten Kontext vertritt – eine Person also durch mehrere andere Personen vertreten wird, die jeweils nur einen bestimmten Teilbereich bearbeiten.

Im Falle der Verwendung von Organisationsmodellen im Process Mining müsste diese Problemstellung in jedem Fall berücksichtigt werden, da eine Fehlinterpretation der Stellvertretung zu fehlerhaften Ableitungen führen wür-

⁸⁰ Rosemann, M.; zur Mühlen, M.: Modellierung der Aufbauorganisation in Workflow-Management-Systemen: Kritische Bestandsaufnahme und Gestaltungsvorschläge, <http://www.b2b-workflow.de/Publications/PDF/MIRO.MIZU-EMISA%281997%29.pdf>, Aburfdatum: 14.06.2010, S. 4.

de. Einer Person würden dabei Rollen zugeordnet werden, die sie in der Realität nicht innehat.

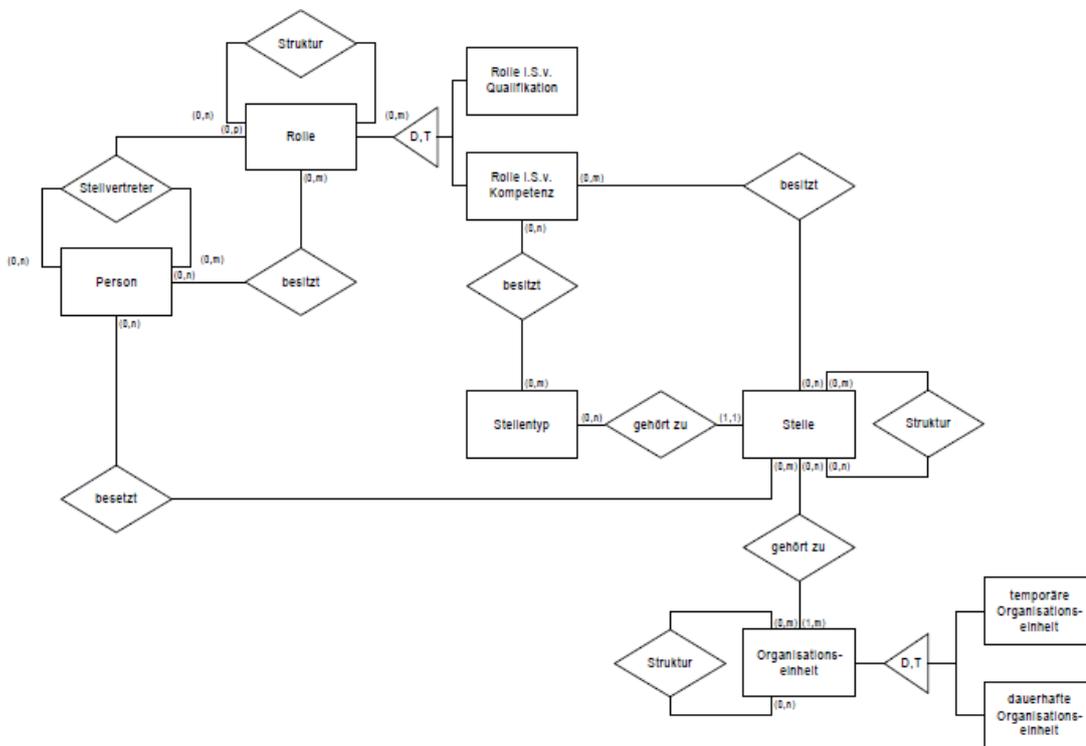


Abb. 16: Referenz-Metamodell der Aufbauorganisation⁸¹

4.3.4 „Staff Assignment Mining“

Die Arbeit von Ly zum Staff Assignment Mining⁸² ist die einzige, die bisher im Bereich der Beziehungen zwischen Aufbauorganisation und Prozess identifiziert werden konnte. Da das Ziel in der Ableitung von Mitarbeiterzuordnungsregeln aus Verlaufsdaten und einem a priori vorhandenen Organisationsmodell liegt, ist der Ansatz in den Bereich des Organizational Mining einzuordnen – es werden organisatorische Informationen extrahiert.

Bearbeiterzuordnungsregeln bestimmen das „Profil“ der zulässigen Mitarbeiter für eine Aktivität, welche sich aus den (minimalen) Qualifikationen zusam-

⁸¹ Rosemann, M.; zur Mühlen, M.: Modellierung der Aufbauorganisation in Workflow-Management-Systemen: Kritische Bestandsaufnahme und Gestaltungsvorschläge, <http://www.b2b-workflow.de/Publications/PDF/MIRO.MIZU-EMISA%281997%29.pdf>, Abrufdatum: 14.06.2010, S. 5, Abb. 1: Referenz-Metamodell der Aufbauorganisation.

⁸² Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 63ff, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

mensetzt, die ein Mitarbeiter erfüllen muss, um die Aktivität ausführen zu können. Um dieses Ziel angehen zu können, setzt der Autor voraus, dass innerhalb der Verlaufsdaten jeder Aktivität genau ein Bearbeiter zugeordnet ist.

Wie bereits das vorherige Kapitel gezeigt hat, verwendet Ly ein Organisations-Metamodell mit den Komponenten Organisationseinheit, Stelle, Rolle, Mitarbeiter und Fähigkeit (Qualifikation) - Hierarchien können unter Rollen und Organisationseinheiten abgebildet werden (siehe Abb. 17). Der Autor hat sich demnach auf die wesentlichen, und für die Erzeugung von Bearbeiterzuordnungsregeln relevanten, Komponenten beschränkt. Es bestehen allerdings wesentliche Definitionsunterschiede, da beispielsweise Rollen maximal genau eine andere Rolle spezialisieren können oder Fähigkeiten sowohl Rollen als auch Mitarbeitern direkt zugeordnet werden können.

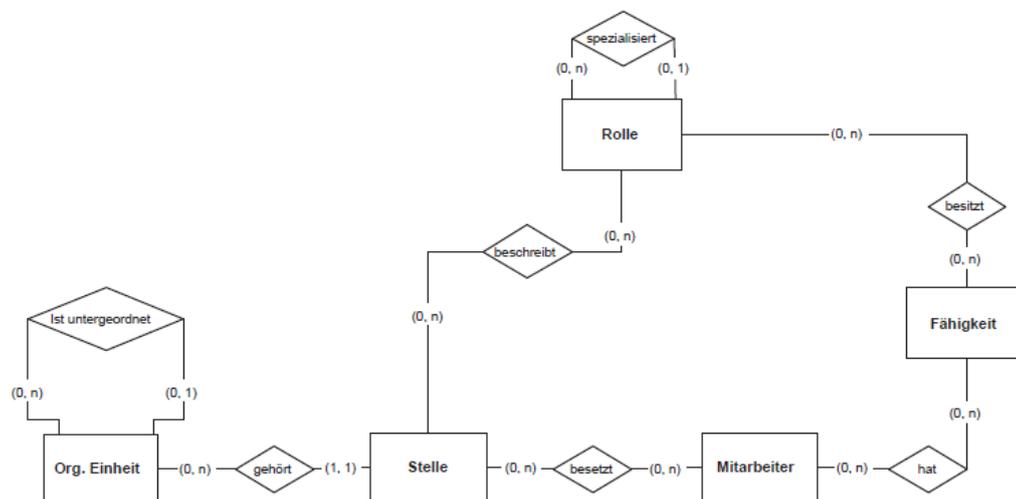


Abb. 17: Verwendetes Organisations-Metamodell im Staff Assignment Mining

Unter Verwendung der Entscheidungsbauminduktion wird nun versucht Regeln abzuleiten, die ein minimales Profil der Bearbeiter einer Aktivität darstellen. Während das Lernproblem beim Mining der Prozesssicht bisher ausschließlich auf positiven Beispielen basiert, sind in diesem Ansatz negative Beispiele absolut natürlich, da für jede Aktivität ermittelt werden muss, ob es sich bei einem Mitarbeiter um einen Bearbeiter oder einen Nicht-Bearbeiter handelt. Angenommen es existieren im Log drei Mitarbeiter (M), die Aktivität A ausführen und über folgende Qualifikationen (Q) verfügen:

- M1: Q1, Q2, Q3, Q4
- M2: Q2, Q4, Q6
- M3: Q2, Q3, Q4, Q5

Da alle Mitarbeiter über die Qualifikationen Q2 und Q4 verfügen, kann diese Menge als minimales Profil betrachtet werden. Allerdings können ebenso wie beim Mining der Prozesssicht unterschiedliche Probleme auftreten, mit denen sich der Autor auseinandersetzt.

Es könnte beispielsweise vorkommen, dass ein Mitarbeiter eine Aktivität nur ausnahmsweise ausführt (z.B. als Vertretung), dadurch jedoch nicht über die Qualifikationen, die zur Ausführung der Aktivität notwendig sind, verfügt. Ebenso könnten Rauschdaten protokolliert werden, welche zu einer fehlerhaften Zuordnung von Mitarbeiter zu Aktivität führen. Zur Problembehandlung werden Häufigkeiten verwendet – es wird davon ausgegangen, dass, wenn ein Mitarbeiter eine Aktivität mit sehr geringer Frequenz ausführt, ein Fehler vorliegen könnte.

Im Falle von negativen Beispielen können keine Frequenzen ermittelt werden, jedoch kann das Beschneiden des Entscheidungsbaumes auf Basis negativer Beispiele sinnvoll sein. Wenn Nicht-Bearbeitern im Organisationsmodell falsche Qualifikationen zugeordnet werden, kann es vorkommen, dass diese schwieriger von Bearbeitern unterschieden werden können. Für diese Fälle wurde ein Grenzwertparameter für negative Beispiele an einem Knoten eingeführt.

Durch die Verwendung eines Organisationsmodells im Mining-Prozess ist dieses, neben den Verlaufsdaten, eine weitere mögliche Fehlerquelle. Wenn beispielsweise für einen regelmäßigen Bearbeiter einer Aktivität eine Qualifikation nicht modelliert wurde, obwohl er diese besitzt, kann die Frequenz nicht mehr als Indiz für einen Ausnahmefall herangezogen werden. Dadurch würden die abgeleiteten Regeln nicht mehr die realen Anforderungen an die Aktivität wiedergeben, sondern die durch das Organisationsmodell implizierte Situation.

Nach Ly können die resultierenden Bearbeiterzuordnungsregeln ggf. durch einen Prozessdesigner nachbearbeitet und in das System überführt werden. Dadurch werden automatisch die passenden Mitarbeiter für eine anstehende Aktivität identifiziert und ausgewählt.

Kritisch sind bei diesem Ansatz mehrere Aspekte anzumerken. Wie in Kapitel 4.3.3 erläutert wurde, beschreibt eine Rolle die für die Ausführung einer Aktivität notwendigen Qualifikationen (und Kompetenzen). Demnach ist im Normalfall die Rolle als Schnittstelle zwischen Aufbau- und Ablauforganisation zu sehen. Da in einer Prozessinstanz jede Aktivität (nach Annahme) jeweils von genau einer Person durchgeführt wird, dient im vorgestellten Ansatz nicht die Rolle, sondern die Person als Schnittstelle zwischen Organisationsmodell und Verlaufsdaten. Das verwendete Organisations-Metamodell lässt weiterhin eine direkte Zuordnung von Fähigkeiten zu einer Person zu, wodurch der Ur-

sprungsgedanke, dass eine Rolle die notwendigen Qualifikationen für die Ausführung einer Aktivität aggregiert, nicht mehr erfüllt ist. Es kann also nicht garantiert werden, dass eine konkrete Rolle, welche für die Erstellung von Bearbeiterzuordnungsregeln sinnvoll wäre, ermittelt werden kann. Vielmehr muss man sich dabei auf die vorhandenen Qualifikationen beziehen. Würde man diese direkte Zuordnung von Fähigkeiten an Mitarbeiter unterbinden und auf die Rollen verlagern, hätte das den Vorteil, dass man die ermittelten (Teil-)Informationen als Extension im Sinne des Workflow Mining verwerten könnten – man könnte also ein vorhandenes Prozessmodell (unabhängig davon, ob es manuell oder durch Process Mining erstellt wurde) mit Rolleninformationen anreichern.

4.3.5 Aktuelle Software

Die Methode zum Staff Assignment Mining wurde nach bisherigem Kenntnisstand noch nicht softwaretechnisch implementiert, jedoch ist eine Umsetzung, beispielsweise als ProM Plug-In, denkbar.

Die Organisationsmodellierung sowie die Kopplung zwischen Aufbau- und Ablauforganisation sind jedoch in zahlreichen Business Process Management Systemen (= BPMS) umgesetzt. Rosemann und zur Mühlen⁸³ nennen beispielsweise FlowMark (IBM), Workparty (SNI), Leu (o.tel.o) und CSE Workflow (CSE). Die Werkzeuge implementieren dabei unterschiedliche Organisations-Metamodelle, wodurch sich auch unterschiedliche Anwendungsszenarien und Funktionalitäten ergeben. Von der Workflow-Unterstützung bis hin zur Organisationsplanung und –gestaltung (hierbei auch die Erstellung von Bearbeiterzuordnungsregeln) werden die Anwendungsszenarien technisch voll unterstützt. Dadurch wird häufig eine feste Kopplung von Aufbau- und Ablauforganisation realisiert, bei welcher sich Änderungen der einen Komponente direkt auf die jeweils andere Komponente auswirken.

⁸³ Rosemann, M.; zur Mühlen, M.: Modellierung der Aufbauorganisation in Workflow-Management-Systemen: Kritische Bestandsaufnahme und Gestaltungsvorschläge, <http://www.b2b-workflow.de/Publications/PDF/MIRO.MIZU-EMISA%281997%29.pdf>, Aburfdatum: 14.06.2010, S. 6ff.

4.4 Ableitung von Organisationsstrukturen aus Verlaufsdaten

Bei der Arbeit von van der Aalst⁸⁴ und Song handelt es sich um die einzige, bei der die Ableitung von Organisationsstrukturen aus Verlaufsdaten angesprochen wird. Sie stellen eine Möglichkeit vor, um Organisationsstrukturen aus Verlaufsdaten zu erraten. Zur Erläuterung wird dazu folgendes Beispiellog angeboten:

Den Daten kann entnommen werden, welcher Mitarbeiter, welche Aktivitäten ausführt. Es kann also festgehalten werden, dass Aktivität A von Tim oder Lisa, Aktivität B und C von Tim, Lisa, Mike oder Carola, Aktivität D von Peter oder Klara und Aktivität E von Klara ausgeführt wird. Aus diesen Informationen können nun Organisationsstrukturen abgeleitet bzw. erraten werden. Es wäre demnach vorstellbar, dass drei Rollen X, Y und Z existieren. X ist dabei für die Ausführung von Aktivität A notwendig, Y für die Ausführung von B und C, Z für die Ausführung von D und E. Damit ergäbe sich folgende Rollenzuordnung:

- **X:** Tim, Lisa
- **Y:** Tim, Lisa, Mike, Carola
- **Z:** Peter, Klara

| Case | Activity | Performer |
|------|----------|-----------|
| 1 | A | Tim |
| 2 | A | Tim |
| 3 | A | Lisa |
| 3 | B | Carola |
| 1 | B | Mike |
| 1 | C | Tim |
| 2 | C | Mike |
| 4 | A | Lisa |
| 2 | B | Tim |
| 2 | D | Peter |
| 5 | A | Lisa |
| 4 | C | Carola |
| 1 | D | Peter |
| 3 | C | Lisa |
| 3 | D | Peter |
| 4 | B | Lisa |
| 5 | E | Klara |
| 5 | D | Klara |
| 4 | D | Peter |

Tabelle 8: Beispiellog - Ableitung von Organisationsstrukturen aus Verlaufsdaten⁸⁵

Diese Erkenntnisse lassen sich wie folgt in einem „Activity-Role-Performer“-Diagramm visualisieren.

⁸⁴ van der Aalst, W.M.P., Song, M. (2004), Mining social networks: uncovering interaction patterns in business processes, in Desel, J., Pernici, B. and Weske, M. (Eds), International Conference on Business Process Management (BPM 2004), Springer Verlag, Berlin, S. 5.

⁸⁵ van der Aalst, W.M.P., Song, M. (2004), Mining social networks: uncovering interaction patterns in business processes, in Desel, J., Pernici, B. and Weske, M. (Eds), International Conference on Business Process Management (BPM 2004), Springer Verlag, Berlin, S. 5.

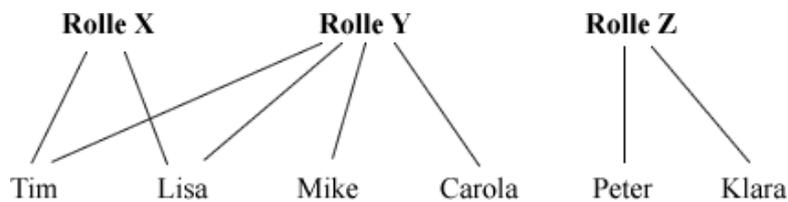


Abb. 18: "Activity-Role-Performer"-Diagramm⁸⁶

Eine detaillierte Ausführung dieser Idee, ebenso wie eine Weiterentwicklung, konnte nicht gefunden werden. Allerdings können verschiedene Probleme bei diesem Vorgehen identifiziert werden – so ist es beispielsweise unklar, wieso nur eine Rolle Z abgeleitet wurde, welche für die Ausführung der Aktivitäten D und E notwendig sein soll, obwohl E ausschließlich von Klara ausgeführt wird. Es ist durchaus vorstellbar, dass Klara zwei Rollen innehat, wobei eine für die Ausführung von Aktivität D und eine für die Ausführung von Aktivität E notwendig ist. Dies wäre auch ein konsistentes Vorgehen zur Literatur, da dort die Rolle häufig als Definition der notwendigen Qualifikationen für die Durchführung einer Aktivität gesehen wird.

Weiterhin stößt das Verfahren schnell an seine Grenzen, wenn Mitarbeiter nur ausnahmsweise eine Aktivität durchführen – beispielsweise als Vertretung eines anderen Mitarbeiters. In diesem Fall kann nicht davon ausgegangen werden, dass der Mitarbeiter über die notwendigen Qualifikationen oder Kompetenzen zur Ausführung der Aktivität verfügt.

Es wurde deutlich, dass es sich bei diesem Ansatz lediglich um eine erste Idee handelt. Es ist jedoch anzumerken, dass andere Autoren, wie beispielsweise Ly⁸⁷, die Aufgabe, Organisationsstrukturen aus Verlaufsdaten abzuleiten, als unrealistisch ansehen. Die Begründung dafür liegt in der Annahme, dass Verlaufsdaten typischerweise nicht genügend Informationen dafür liefern.

4.5 Ableitung von Prozesswissen unter der Verwendung von Verlaufsdaten und Organisationswissen

Im Bereich der Ableitung von Prozesswissen unter der Verwendung von Verlaufsdaten und Organisationswissen sind bisher keine Arbeiten bekannt. Es liegt jedoch die Annahme zugrunde, dass Organisationswissen wertvolle Informationen für das Process Mining liefern könnte. Dieser Gedanke wird in

⁸⁶ van der Aalst, W.M.P., Song, M. (2004), Mining social networks: uncovering interaction patterns in business processes, in Desel, J., Pernici, B. and Weske, M. (Eds), International Conference on Business Process Management (BPM 2004), Springer Verlag, Berlin, S. 6.

⁸⁷ Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 57, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

Kapitel 5 bei der Entwicklung der Mining Methode weiter verfolgt und konkretisiert.

Voraussetzung für die Verwendung von Organisationswissen im Process Mining ist eine Kopplung eines a priori vorhandenen Organisationsmodells mit den Verlaufsdaten, ähnlich wie sie von Ly in seinem Ansatz zum Staff Assignment Mining vorgeschlagen wird – dabei sollte jedoch das Organisation-Metamodell grundsätzlich überdacht werden.

Ein solcher Ansatz hat demnach einen engen Bezug zum hier dargelegten Aspekt der Beziehungen zwischen Aufbau- und Ablauforganisation (Kapitel 4.3), ebenso wie zum Process Mining im Allgemeinen (Kapitel 3).

4.6 Zusammenfassung und Fazit

Das Kapitel hat den aktuellen Forschungsstand und das Zusammenspiel von Process Mining und Organisationswissen aufgezeigt. Entsprechend der ursprünglichen Erkenntnis, dass organisatorischen Aspekten im Process Mining bisher wenig Bedeutung zukam, konnte gezeigt werden, dass sich die bisherigen Forschungsanstrengungen in diesem Bereich in den Anfängen befinden.

Es haben sich hierbei zwei Teilgebiete, die Beziehungen zwischen Akteuren und die Beziehungen zwischen Aufbau- und Ablauforganisation, herauskristallisiert, welche in vergangenen Arbeiten konkret thematisiert wurden. Auf dem Gebiet der Beziehungen zwischen Akteuren wurde exemplarisch die Arbeit „Mining Social Networks“⁸⁸ vorgestellt und kritisch betrachtet. Die beiden anderen Arbeiten von Ang et. al.⁸⁹ und van der Aalst et. al.⁹⁰ sind dieser im Hinblick auf die methodische Umsetzung sehr ähnlich (basieren ebenfalls auf speziellen Metriken), sodass auf eine detaillierte Vorstellung verzichtet wurde.

Das Teilgebiet der Beziehungen zwischen Aufbau- und Ablauforganisation stellte sich als wesentlich umfangreicher heraus, da verschiedene Aspekte bereits in der Vergangenheit umfangreich thematisiert wurden. Beispielsweise beschäftigten sich einige Autoren, unabhängig vom Process Mining, mit den

⁸⁸ van der Aalst, W.M.P., Song, M. (2004), Mining social networks: uncovering interaction patterns in business processes, in Desel, J., Pernici, B. and Weske, M. (Eds), International Conference on Business Process Management (BPM 2004), Springer Verlag, Berlin.

⁸⁹ Gao, A., Yang, Y., Zeng, M., Zhang, J., Wang, Y.. Organizational Structure Mining Based on Workflow Logs, In: International Conference on Business Intelligence and Financial Engineering 2009 (BIFE '09).

⁹⁰ van der Aalst, W.M.P; Reijers, H.A.; Song, M.: Discovering Social Networks from Event-Logs, In: Computer Supported Cooperative Work (CSCW) Volume 14, Number 6, 549-593.

Potentialen von Organisationswissen, mit Organisations-Metamodellen und insbesondere mit der Kopplung von Aufbau- und Ablauforganisation. Bezogen auf das Process Mining konnte eine konkrete Methode, das „Staff Assignment Mining“⁹¹, identifiziert werden, welche vorgestellt und in den wesentlichen Zügen erläutert wurde.

Als weiteres Teilgebiet konnte die Ableitung von Prozesswissen unter der Verwendung von Organisationswissen und Verlaufsdaten identifiziert werden – in diesem Bereich sind bisher jedoch keine Arbeiten, außer der vorliegenden, bekannt.

Während die Beziehungen zwischen Akteuren und die Beziehungen zwischen Aufbauorganisation und Prozess zum Organizational Mining (Process Mining in der Organisationssicht) zu zählen sind, ist das zuletzt genannte Teilgebiet in das Control-Flow Mining (Process Mining in der Prozesssicht) einzuordnen.

Auf Basis der Ausführungen aus Kapitel 3 und 4 soll nun in den folgenden Kapiteln eine neue Methode zum Process Mining unter der Verwendung von Organisationswissen und Verlaufsdaten erarbeitet und prototypisch implementiert werden.

⁹¹ Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 63ff, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

5 Entwicklung einer Methode zum Process Mining unter der Verwendung von Organisationswissen

5.1 Einführung und Überblick

5.1.1 Zielsetzung

Das Ziel dieses Kapitels liegt in der Entwicklung einer neuen Methode zum Process Mining in der Prozesssicht. Im Gegensatz zu allen bisher bekannten Ansätzen in diesem Bereich soll neben den Verlaufsdaten der angebotenen IT-Systeme auch Organisationswissen in den Mining Prozess einbezogen werden. Diese Idee basiert auf der Annahme, dass Organisationswissen wertvolle Informationen für die Behandlung unterschiedlicher Problemstellungen im Process Mining (siehe Kapitel 3.6) liefern kann.

Die entwickelte Methode soll anschließend in Form eines ProM-Plugins prototypisch implementiert werden.

5.1.2 Anforderungen

Wie bereits angesprochen, soll die Methode dem Bereich der Prozesssicht zugeordnet werden können – der Output soll demnach einen Prozess repräsentieren. Es muss zwischen Anforderungen an die Input-Daten, Anforderungen an den Algorithmus und Anforderungen an die Output-Daten unterschieden werden.

Als Input-Daten sollen einerseits, wie im Process Mining üblich, die Verlaufsdaten der angebotenen IT-Systeme herangezogen werden. Im Hinblick auf die zu entwickelnde Methode sollen diese Verlaufsdaten möglichst minimalen Anforderungen genügen, sodass nur die Informationen gefordert werden, die für den Algorithmus relevant sind (siehe hierzu Kapitel 3.4). Andererseits soll ein Organisationsmodell als Input dienen, welches die für den Algorithmus notwendigen Informationen bereitstellt und eine Kopplung von Aufbauorganisation und Verlaufsdaten ermöglicht. Mit der geplanten prototypischen Realisierung als ProM-Plugin muss weiterhin geklärt werden, wie die Input-Daten aufbereitet werden müssen, um im Prototyp verwendet werden zu können (Kapitel 6).

Für den Algorithmus sollen keine besonderen Restriktionen bzgl. der verwendeten Methoden gestellt werden – es können also sowohl algorithmische, statische als auch hybride Ansätze verwendet werden. Diese Ansätze können durchaus auf bewährten Ansätzen aufbauen. Eine Pre-Processing Phase soll im Vorfeld die Verlaufsdaten bereinigen, und die Kopplung von Organisations-

modell und Verlaufsdaten realisieren. In einem eventuellen Post-Processing soll das resultierende Modell verfeinert werden.

Der Output soll den abgeleiteten Prozess grafisch repräsentieren.

5.1.3 Herangehensweise

Zu Beginn soll auf die atomare Betrachtung von Aktivitäten als Problemstellungen im Process Mining eingegangen und Alternativen aufgezeigt werden.

Entsprechend der Anforderungen gilt es mehrere Problemstellungen zu bearbeiten. Zunächst muss geklärt werden, welche Potentiale Organisationswissen darstellen, die den Mining-Prozess unterstützen können. Hierzu sollen die unterschiedlichen Problemstellungen im Process Mining (siehe Kapitel 3.6) einzeln betrachtet und ein eventueller Bezug zu organisatorischen Aspekten hergestellt werden. Dadurch sollen erste nicht-formale Ideen für die Einbeziehung des Organisationswissens in den Mining-Prozess gesammelt und festgehalten werden. In dieser Phase spielen die in Kapitel 4.3.2 vorgestellten allgemeinen Potentiale des Organisationswissens eine weitere signifikante Rolle, da auch diese für die Ableitung von Ideen zum Process Mining interessanten Input liefern können.

Im Anschluss sollen die gesammelten Ideen genutzt werden, um Anforderungen an das Organisationswissen zu definieren. Zusammen mit den Ausführungen in Kapitel 4.3.3 soll ein Organisations-Metamodell erarbeitet werden, welches einerseits den State-of-the-Art der Organisations-Metamodelle repräsentiert, andererseits auf die gestellten Anforderungen abgestimmt ist.

Nun sollte die Form der grafischen Repräsentierung des Zielmodells ausgewählt werden. Das ist bereits an dieser Stelle wichtig, da der eigentliche Mining-Prozess Informationen für das Modell generieren muss. Wird beispielsweise die eEPK als Zielmodell ausgewählt, könnten Informationen über Organisationseinheiten erzeugt werden, welche bei einem Petri-Netz nicht annotierbar wären.

Basierend auf dem gewählten Organisations-Metamodell, den ausgearbeiteten Anforderungen an die Verlaufsdaten und dem gewählten Zielmodell, sollen im Anschluss die gesammelten Ideen formalisiert werden.

5.2 Atomare Betrachtung von Aktivitäten als Problemstellung im Process Mining

Nahezu alle bisherigen Ansätze zum Process Mining fordern in den Verlaufsdaten eine einmalige Darstellung jeder Aktivität, wobei sich der zugehörige Zeitstempel auf den Abschluss dieser Aktivität bezieht. Diese Anforderung impliziert eine atomare Betrachtung von Aktivitäten und es kann nicht nachvollzogen werden, ob verschiedene Aktivitäten zeitlich überlappen. Diese Betrachtungsweise ist vor allem auch deswegen kritikwürdig, da es durchaus denkbar ist, dass ein Mitarbeiter mehrere verschiedene Aktivitäten „gleichzeitig“ ausführt, obwohl er zu einem bestimmten Zeitpunkt nur mit einer Aktivität aktiv beschäftigt sein kann.

Angenommen ein Mitarbeiter eines Softwareherstellers befasst sich mit der Dokumentation von Entwicklungs-Projekten. Da es unwahrscheinlich ist, dass dieser Mitarbeiter alle relevanten Informationen für seine Dokumentation bei Projektabschluss rekonstruieren kann, wird dieser bereits in einer sehr frühen Projektphase mit der Dokumentation beginnen. Da die Dokumentation den gesamten Projektverlauf widerspiegeln soll, kann sie erst mit Abschluss des Projektes fertiggestellt werden. Für die Verlaufsdaten bedeutet das, dass die Aktivität immer sehr weit hinten in den Prozessinstanzen auftaucht, woraus geschlossen wird, dass es sich bei „Prozess dokumentieren“ um eine der letzten Aktivitäten des Entwicklungs-Prozesses handelt. In der Realität läuft diese Aktivität allerdings parallel zu anderen Aktivitäten ab, sodass das abgeleitete Prozessmodell fehlerhaft wäre.

Da davon auszugehen ist, dass die Mehrzahl der Informationssysteme nicht nur den Endzeitpunkt einer Aktivität, sondern auch den Startpunkt protokollieren, könnte eine Lösung in der Verwendung dieser Information liegen. Dies hätte einerseits den Vorteil, dass detailliertere Performance-Daten aus dem Mining-Prozess resultieren und andererseits der Mining-Prozess selbst erheblich unterstützt würde. Statt, wie bisher, Metriken als Indikator für Parallelität heranzuziehen, könnte diese auf Basis von „Überlappungen“ in den Verlaufsdaten sehr einfach abgeleitet werden. Auch kompliziertere Strukturen, wie oben beschrieben (eine Aktivität läuft parallel zu vielen anderen Aktivitäten ab) könnten ebenso einfach erkannt werden.

5.3 Organisationswissen im Process Mining

5.3.1 Potentiale

5.3.1.1 Doppelte Tasks und Rollen-Annotation

Organisationswissen kann entscheidende Beiträge für die Erkennung doppelter Tasks (siehe Kapitel 3.6) im Process Mining liefern. Jede Aufgabe innerhalb eines Prozesses stellt definierte Anforderungen an die ausführenden Mitarbeiter, welche in einem Organisationsmodell als Qualifikationen festgehalten werden. Die Qualifikationen werden dann bestimmten Rollen oder den Mitarbeitern direkt zugeordnet. Ersteres ist hierbei von besonderer Relevanz, da entsprechend des State-of-the-Art der Kopplung von Aufbau- und Ablauforganisation davon auszugehen ist, dass die für die Ausführung einer Aufgabe erforderlichen Qualifikationen in einer Rolle gebündelt werden.

Die minimale Menge an Qualifikationen (Rollen) die für die Ausführung einer Aktivität notwendig ist, könnte demnach ermittelt werden, indem die Schnittmenge der Qualifikationen (Rollen) der ausführenden Mitarbeiter berechnet wird. Angenommen die Aktivität A wird nach den Verlaufsdaten stets von den beiden Mitarbeiter M1 und M2 ausgeführt, wobei M1 die Rollen R1 und R2 innehat, M2 die Rollen R2 und R3. Daraus kann geschlossen werden, dass R2 für die Ausführung von A obligatorisch ist. R1 und R3 kommen nicht in Betracht, da nur jeweils einer der beiden Mitarbeiter über eine dieser Rollen verfügt. Man gewinnt also Informationen darüber, welche Rollen für die Ausführung einer Aktivität notwendig sind, sodass eine Kopplung von Aufbau- und Ablauforganisation im Zielmodell durch die Annotation der Rollen realisiert werden kann.

Angenommen es tritt nun der Fall ein, dass M2 nicht über die Rollen R2 und R3, sondern über die Rollen R3 und R4 verfügt – es ergäbe sich eine leere Rollen-Schnittmenge, da die Rollen von M1 und M2 vollkommen unterschiedlich sind. Tritt dieser Fall häufig auf, kann davon ausgegangen werden, dass es sich bei der im Log dargestellten Aktivität A, in Wirklichkeit um zwei unterschiedliche Aktivitäten A1 und A2 handelt, da Mitarbeiter mit überschneidungsfreien Qualifikationen faktisch nicht dieselben Aufgaben ausführen können.

Sollte hierbei mit dem Rollenkonstrukt gearbeitet werden (statt direkt mit konkreten Qualifikationen) muss jedoch noch geprüft werden, ob die Qualifikationen, die über die Rollen aggregiert werden, überschneidungsfrei sind. Es ist denkbar, dass unterschiedliche Rollen gleiche Qualifikationen beinhalten. Man würde in diesem Fall allerdings von der Forderung abweichen, dass die notwendigen Qualifikationen für die Ausführung einer Aktivität in einer Rolle

zusammengefasst werden. Dies macht deutlich, dass die Definition des Organisations-Metamodells, sowie Annahmen über die Kopplung von Aufbau- und Ablauforganisation ausschlaggebend für das Vorgehen im Process Mining sind.

Dieser Ansatz ist auch deshalb sehr wichtig, da er bereits im Pre-Processing denkbar ist. So könnten doppelte Tasks im Vorfeld erkannt und bereinigt werden. Anschließend kann eine beliebige Methode das eigentliche Mining übernehmen.

5.3.1.2 Parallele / Konkurrierende Abläufe

In bestimmten Konstellationen kann die im Log vorhandene Information über den Initiator einer Aktivität Informationen darüber liefern, ob Parallelität vorliegt oder nicht. Angenommen im Log existieren zwei Aktivitäten D und E, welche entweder in der Reihenfolge DE oder ED ausgeführt werden. In diesem Fall stellt sich die Frage, ob die beiden Aktivitäten parallel ausgeführt werden können oder konkurrieren.

Wird bei der Betrachtung der Initiator-Information festgestellt, dass sowohl D als auch E immer von Mitarbeiter M1 ausgeführt werden, dann kann daraus geschlossen werden, dass die Aktivitäten nicht parallel ausgeführt werden können. Bei dieser Schlussfolgerung ist jedoch die Annahme zu beachten, dass ein Mitarbeiter zu einem Zeitpunkt immer nur genau einer Aktivität nachgehen kann – Aktivitäten werden also atomar betrachtet.

Berücksichtigt man Kapitel 5.2, welches sich mit der Nicht-atomaren Betrachtung von Aktivitäten beschäftigt, ist dieser Ansatz nicht mehr haltbar. Es ist dabei möglich, dass eine Person zu einem konkreten Zeitpunkt mehr als einer Aktivität nachgeht. Es ist weiterhin denkbar, dass eine Aktivität zugunsten einer anderen Aktivität unterbrochen wird, was in den Verlaufsdaten jedoch nicht abgebildet wird. Grafisch ließe sich das wie folgt beschreiben.

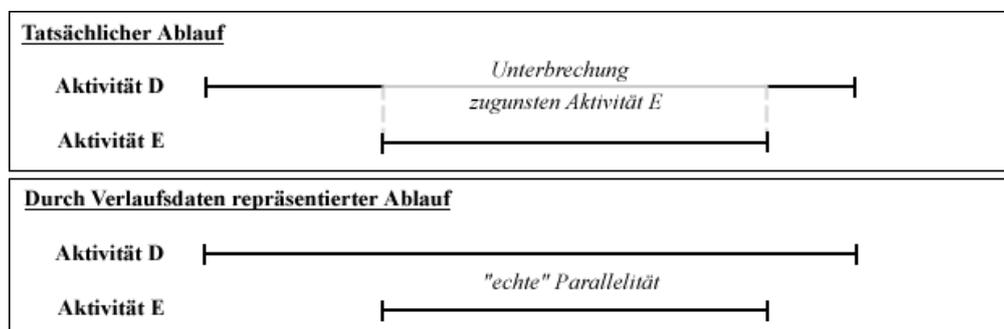


Abb. 19: Tatsächliche und durch Log repräsentierte Parallelität

In diesem Fall müsste zunächst definiert werden, wie Parallelität zu verstehen ist. Im oberen Fall (Abb. 19) befasst sich der Mitarbeiter zwar mit beiden Aktivitäten „zugleich“, allerdings arbeitet er zu jedem konkreten Zeitpunkt an genau einer Aufgabe. Der untere Fall spiegelt einen echten parallelen Ablauf wieder, da sich der Mitarbeiter zu jedem konkreten Zeitpunkt in der Zeitspanne von Aktivität E mit zwei Aktivitäten aktiv befasst. Dieser Fall könnte beispielsweise eintreten, wenn sich ein Unternehmensberater auf der Anfahrt zum Kunden befindet und gleichzeitig mit diesem Kunden telefonisch das weitere Vorgehen abstimmt. Ein Beispiel für die Unterbrechung zugunsten einer anderen Aktivität könnte durch Weiterführung des Projekt-Dokumentations-Beispiels aus Kapitel 5.2 dargestellt werden. Es ist sehr unwahrscheinlich, dass der ausführende Mitarbeiter, während er aktiv mit der Erstellung der Dokumentation beschäftigt ist, einer weiteren Tätigkeit nachgeht. Andererseits wird ihn diese Aktivität nicht über den gesamten Projektverlauf auslasten, sodass er diese diverse Male zugunsten anderer Aktivitäten unterbrechen wird.

Ein Ansatz für die Unterscheidung dieser beiden Fälle könnte wiederum auf dem Organisationswissen basieren. Würde das Organisationsmodell Informationen über den Ausführungsort von Aktivitäten enthalten, könnte man in bestimmten Fällen eine echte Parallelität leicht ausschließen. Betrachtet man den Fall aus Abb. 19, so würde man die Ausführungsorte von D und E vergleichen und beispielsweise feststellen, dass D in Köln und E in Hamburg ausgeführt wird. Man kann mit Sicherheit ableiten, dass D und E nicht echt parallel ablaufen, da sich eine Person zu jedem Zeitpunkt nur an genau einem Ort befinden kann.

Vorstellbar wäre die Modellierung der Ortsinformation als Attribut an der Rolle oder als eigene Entität mit einer 1:1-Beziehung zur Rolle. Voraussetzung ist wiederum, dass die Qualifikationen zur Ausführung einer Aktivität konsequent in jeweils genau einer Rolle gebündelt werden und diese eine Rolle vom Mining-Algorithmus abgeleitet werden kann.

5.3.1.3 *Rauschen*

Bei der Verwendung von Organisationswissen im Process Mining ergeben sich bei der Betrachtung der Rauschdaten erschwerte Bedingungen – Rauschen kann nicht mehr nur in den Verlaufsdaten, sondern auch im Organisationsmodell auftreten. Während es sich jedoch beim Rauschen in den Verlaufsdaten um Fehler bei der Protokollierung durch ein Informationssystem handelt, handelt es sich beim Rauschen im Organisationsmodell um Fehler bei Modellierung durch menschliche oder maschinelle Ressourcen. Es könnte beispielsweise der

Fall eintreten, dass Qualifikationen oder Rollen falsch oder gar nicht modelliert werden, was während des Mining Prozesses, beispielsweise bei der Erkennung doppelter Tasks, zu Fehlern führen würde. Deshalb ist die Korrektheit des Organisationsmodells ausschlaggebend für die Qualität und die Korrektheit des Output-Modells.

Trotz dieser erschwerten Bedingungen kann Organisationswissen einen Beitrag zur Erkennung von Rauschdaten (sowohl im Log als auch im Organisationsmodell) leisten. Angenommen in einem Log existiert die Aktivität A, welche stets von den Mitarbeitern M1 und M2 ausgeführt wird. Die Schnittmenge der Rollen der beiden Mitarbeiter sei $\{R2\}$, sodass gemäß Kapitel 5.3.1.1 davon auszugehen ist, dass R2 für die Ausführung von A benötigt wird (R2 aggregiert alle für die Ausführung von A notwendigen Qualifikationen). Nun taucht im Log als Einzelfall (oder sehr selten) die Aktivität A mit dem Initiator M3 auf, dessen Rollenschnittmenge mit $\{R2\}$ die leere Menge $\{\}$ ist. D.h. M3 verfügt nicht über die Rolle R2, sodass hier zwei Fälle vorstellbar sind. Es kann sich einerseits um einen doppelten Task, wie in Kapitel 5.3.1.1 beschrieben, handeln, andererseits um Rauschen. Da M3 die Aktivität A nach den Verlaufsdaten nur einmal (bzw. sehr selten) ausführt, ist Zweiteres eher denkbar. Demnach verfügt M3 nicht über die Qualifikationen, um A ausführen zu können, was wiederum zu zwei möglichen Ursachen führen kann – es handelt sich um einen Fehler in den Verlaufsdaten oder um einen Fehler im Organisationsmodell. Ein Fehler in den Verlaufsdaten wäre, wie bereits erwähnt, auf eine fehlerhafte Protokollierung zurückzuführen, ein Fehler im Organisationsmodell auf eine fehlerhafte Modellierung. Bei der fehlerhaften Modellierung könnte entweder eine Zuordnung der Rolle R2 zu M3 vergessen worden sein, oder es handelt sich bei M3 um den Stellvertreter von M1 oder M2, wobei die entsprechende Verbindung unterschlagen wurde.

Es wäre vorstellbar diese Rauschdatenerkennung über einen Parameter zu steuern, welcher als Grenzwert eine Unterscheidung zwischen Rauschen und doppelten Tasks vornimmt. Solche Schwellenwerte wurden bereits erfolgreich, beispielsweise beim Markov-basierten Process Mining⁹² oder dem Staff Assignment Mining,⁹³ eingesetzt.

⁹² Cook, J.E.; Wolf, A.L.: Discovering models of software processes from event-based data, In: ACM Transactions on Software Engineering and Methodology, Vol. 7 No. 3, S. 233.

⁹³ Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 63ff, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

5.3.1.4 *Bearbeiterzuordnungsregeln*

Ly entwirft in seiner Arbeit zum Staff Assignment Mining⁹⁴ eine Methode, mit welcher Bearbeiterzuordnungsregeln aus Verlaufsdaten und einem Organisationsmodell abgeleitet werden können. Obwohl dies in den Bereich des Organizational Mining einzuordnen ist, bestehen durchaus Synergieeffekte zur Prozesssicht.

Eine vorgelagerte Bereinigung der Verlaufsdaten im Sinne von doppelten Tasks (siehe Kapitel 5.3.1.1) kann die Methode optimieren. Diese Bereinigung könnte als Pre-Processing Phase integriert werden, sodass doppelte Tasks im Vorfeld identifiziert und dadurch entsprechende Umbenennungen vorgenommen werden können.

Ebenso wäre es denkbar, eine Ableitung von Bearbeiterzuordnungsregeln im Anschluss an das Mining der Prozesssicht durchzuführen. Wenn der Methode zur Ableitung von Bearbeiterzuordnungsregeln statt der Verlaufsdaten ein Prozessmodell zugrunde liegen würde, könnten über die bisherigen Output-Informationen hinaus zusätzliche Zuordnungsbedingungen (engl. „Conditions“) generiert werden. Angenommen es existieren im Log die Aktivitäten A und B, welche ausschließlich von den Mitarbeiter M1 und M2 ausgeführt werden und echt parallel ablaufen können (was nach dem Mining der Prozesssicht entsprechend Kapitel 5.3.1.2 bekannt ist). Standardmäßig würde eine Bearbeiterzuordnungsregel eine anstehende Aktivität einem „freien“ Mitarbeiter zuweisen. Wenn nun beide Mitarbeiter bereits mit einer Aufgabe befasst sind, würde daraus ein Wartestatus resultieren. Ist die Kenntnis über die Parallelität bekannt, könnte dies in die Regel aufgenommen werden, sodass trotz Belegung beider Mitarbeiter eine Zuordnung erfolgt.

5.3.1.5 *Prozessablauf in der Organisation*

Eine weitere Möglichkeit, um Nutzen aus Organisationswissen im Process Mining zu ziehen, liegt in der Abbildung des Prozessverlaufs in der Organisationsstruktur. Generell ist es interessant zu wissen, welche Organisationseinheiten bzw. Abteilung innerhalb eines Unternehmens an welchem Prozess beteiligt sind. Dies kann einerseits durch die Annotation von Organisationseinheiten im Prozessmodell selbst (z.B. eEPK) geschehen, andererseits können die beteiligten Organisationseinheiten zusammengefasst und übersichtlich, beispielsweise über das Organisationsmodell, dargestellt werden.

⁹⁴ Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 63ff, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

In herkömmlichen Ansätzen zum Process Mining, bei welchen kein Organisationswissen vorhanden ist, können lediglich die Initiatoren am Prozessmodell annotiert werden. Wenn jedoch a priori ein Organisationsmodell vorliegt, könnten die Initiatoren leicht auf ihre Organisationseinheiten abgebildet werden, sodass diese Aufgabe auf einfache Weise realisierbar ist.

Man erhält so einen Überblick über die Aufgaben von Organisationseinheiten, welche vom Unternehmen für Planungen unterschiedlichster Art (beispielsweise für die Definition von Mitarbeiteranforderungen in der Mitarbeiterakquise und -entwicklung) genutzt werden können.

5.3.1.6 Prozessidentifikation und -separierung

Typischerweise wird in Unternehmen eine Vielzahl von Prozessen ausgeführt, welche sich in den Verlaufsdaten niederschlagen – so ist es häufig der Fall, dass die exportierten Verlaufsdaten nicht nur einen, sondern mehrere unterschiedliche Prozesse beinhalten, welche identifiziert und separiert werden müssen. Würde eine solche Separierung nicht vorgenommen werden, hätte dies ein sehr umfangreiches und komplexes Output-Modell zur Folge.

Eine Methode, welche sich mit dieser Problematik auseinandersetzt, ist beispielsweise das Trace Clustering⁹⁵. Dabei werden Prozessinstanzen auf die beteiligten Aktivitäten und deren Ausführungsreihenfolge hin untersucht. Diese Informationen werden dann zur Gruppierung von Aktivitäten zu einem (Teil)-Prozess genutzt, was einer Aufteilung in mehrere Prozesse entspricht (engl. „clustering“).

Eine Grenze solcher Ansätze besteht insbesondere bei sehr ähnlichen Prozessen, welche sich in wenigen Aktivitäten und Abläufen unterscheiden. Wenn die Unterschiede gering sind, kann das herkömmliche Clustering nicht greifen, sodass keine Separierung des Prozesses vorgenommen wird und im Zweifel mehrere Prozesse aggregiert dargestellt werden. Ein weiteres Problem liegt in der Tatsache, dass das Clustering auf Basis von Algorithmen durchgeführt wird, und damit ein korrektes Clustering nicht garantiert werden kann. Allein die Tatsache, dass Prozessabläufe sehr unterschiedlich sind, garantiert nicht, dass es sich um unterschiedliche Prozesse handelt.

Auch hier kann Organisationswissen einen Beitrag leisten. Angenommen ein Vertriebsunternehmen führt im Rahmen seiner Auftragsakquise eine Kundensegmentierung durch, sodass ein Teil der Mitarbeiter ausschließlich Privatkun-

⁹⁵ Song, M., Günther, C.W., van der Aalst, W.M.P.: Trace Clustering in Process Mining, In: In Proceedings of Business Process Management Workshops 2008.

den, der andere Teil der Mitarbeiter ausschließlich Geschäftskunden bedient. Es ergeben sich dadurch auch unterschiedliche Vorgehensweisen, da davon auszugehen ist, dass Geschäftskunden mehr Umsatz generieren als Privatkunden. Die beiden Prozesse könnten wie folgt aussehen.

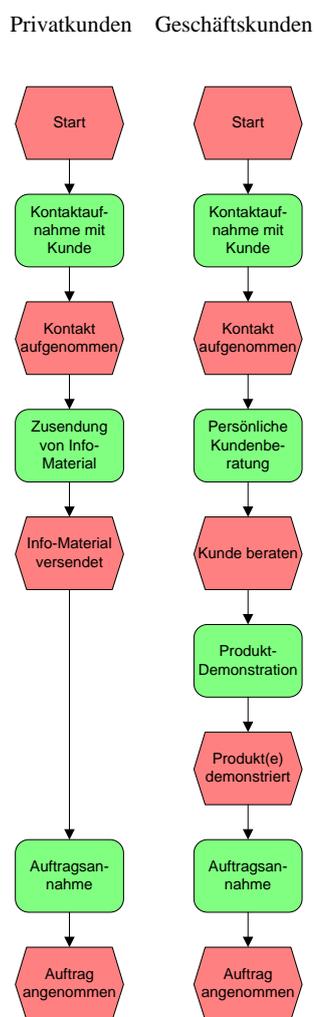


Abb. 20: Kundensegmentierung (EPK)

Interessant ist dieser Ansatz insbesondere aus zwei Gründen. Einerseits sind sehr wenig Prozessinstanzen für das „Clustering“ notwendig, andererseits kann die Korrektheit garantiert werden, da a priori bekannt ist, dass Geschäftskundenberater keine Privatkunden bedienen und umgekehrt. Demnach muss es sich um unterschiedliche Prozesse handeln. Voraussetzung für die Korrektheit der Prozess-Separierung ist allerdings die Korrektheit der Verlaufsdaten und des Organisationswissens. Die Realisierung der Zuständigkeit von einzelnen

Der wesentliche Unterschied besteht demnach darin, dass den Privatkunden Informationsmaterial zugesendet wird, während Geschäftskunden eine persönliche Beratung und eine Produktdemonstration erhalten.

Aus entsprechenden Verlaufsdaten würden herkömmliche Verfahren einen Prozess ableiten, welcher nach dem Ereignis „Kontakt aufgenommen“ ein XOR-Konstrukt enthält, auf welches die beiden Pfade „Zusendung von Info-Material“ (1) und „Persönliche Kundenberatung“ → „Produktdemonstration“ (2) folgen würden.

Hier kann Organisationswissen das Clustering erheblich verbessern. Angenommen eine Prozessinstanz innerhalb der Verlaufsdaten enthält das Ereignis „Persönliche Kundenberatung“, welches von Mitarbeiter M1 ausgeführt wird. Eine weitere Instanz beinhaltet die Aktivität „Zusendung von Info-Material“, welche von M2 ausgeführt wird. Wenn aus dem Organisationswissen bekannt ist, dass M1 für den Geschäftskundensektor und M2 für den Privatkundensektor zuständig ist, können die beiden Instanzen sofort einem bestimmten Segment zugeordnet werden, was zu einer Separierung der beiden Instanzen in zwei unterschiedliche Prozesse führt.

Mitarbeitern wäre im konkreten Fall über deren Rollen bzw. Qualifikationen / Kompetenzen denkbar.

5.3.2 Anforderungen an das Organisationsmodell

Die folgenden Anforderungen ergeben sich aus den vorgestellten Potentialen des Organisationswissens für das Process Mining aus dem vorherigen Kapitel.

Es wurde darin wiederholt angemerkt, dass es sinnvoll erscheint, die für die Ausführung von Aktivitäten notwendigen Qualifikationen in Rollen zu bündeln. Auf die zusätzliche Modellierung von Kompetenzen kann verzichtet werden, wenn diese in Form von Qualifikationen abgebildet werden können. Dies ist vor allem auch auf Kapitel 4.3.3 (Organisations-Metamodelle) zurückzuführen, welches gezeigt hat, dass Kompetenzen in wenigen der untersuchten Metamodellen zu finden waren, während Qualifikationen fast durchgängig modelliert wurden. Da Qualifikationen in der genannten Form gebündelt werden, ist weiterhin keine Zuordnung von Qualifikationen zu Personen (Mitarbeitern) erforderlich, da diese keine Auswirkungen auf die auszuführenden Tätigkeiten hätten. Wenn ein Mitarbeiter nur über einen Teil der für die Ausführung einer Aktivität notwendigen Qualifikationen verfügt, wird er diese trotzdem nicht ausführen können.

Angenommen ein Mitarbeiter M1 verfügt über die Rollen R1 und R2, jedoch nicht über Rolle R3. Jede Rolle beinhaltet die für die Ausführung der Aktivität A1-A3 notwendigen Qualifikationen. R1 aggregiert beispielsweise Q1 und Q2, R2 aggregiert Q3, Q4 und Q5, R3 aggregiert Q1, Q3 und Q4. Dadurch würde M1 zwar über alle notwendigen Qualifikationen zur Ausführung von A3 verfügen, da er jedoch nicht die Rolle R3 innehat soll davon ausgegangen werden, dass er diese nicht ausführen wird.

Aus Kapitel 5.3.1.2 (Parallele / konkurrierende Abläufe) ergibt sich weiterhin die Forderung nach der Modellierung des Ausführungsortes einer Aktivität. Hierbei soll einerseits eine Verbindung zur Rolle existieren, welche nunmehr als Äquivalent der Aufbauorganisation zur Aktivität in der Ablauforganisation zu verstehen ist, andererseits ist eine Kopplung an die Person (bzw. Mitarbeiter) notwendig, da es denkbar ist, dass Aktivitäten an unterschiedlichen Standorten im Unternehmen ausgeführt werden.

Es sei zusätzlich darauf hingewiesen, dass die Qualität des Organisationsmodells, also Vollständigkeit und Korrektheit, ausschlaggebend für die Ergebnisse des Mining-Prozesses sind. Wenn beispielsweise die Zuordnung einer Rolle zu einem Mitarbeiter im Organisationsmodell nicht existiert, kann die Ableitung

von Rauschen (siehe Kapitel 5.3.1.3), ebenso wie die Erfassung von doppelten Tasks (siehe Kapitel 5.3.1.3) und weitere Aspekte zu fehlerhaften Ergebnissen führen.

5.3.3 Organisations-Metamodell

Basierend auf den gestellten Anforderungen und anlehnend an Kapitel 4.3.3 wurde das nachfolgende Organisations-Metamodell erstellt. Es enthält alle Konstrukte und Verbindungen, welche für die Umsetzung der in Kapitel 5.3.1 vorgestellten Ansätze notwendig sind, sodass es für die hier zu entwickelnde Mining-Methode herangezogen werden kann. Ebenso sind alle wichtigen organisatorischen Konstrukte enthalten, sodass ein enger Zusammenhang zum State-of-the-Art gegeben ist.

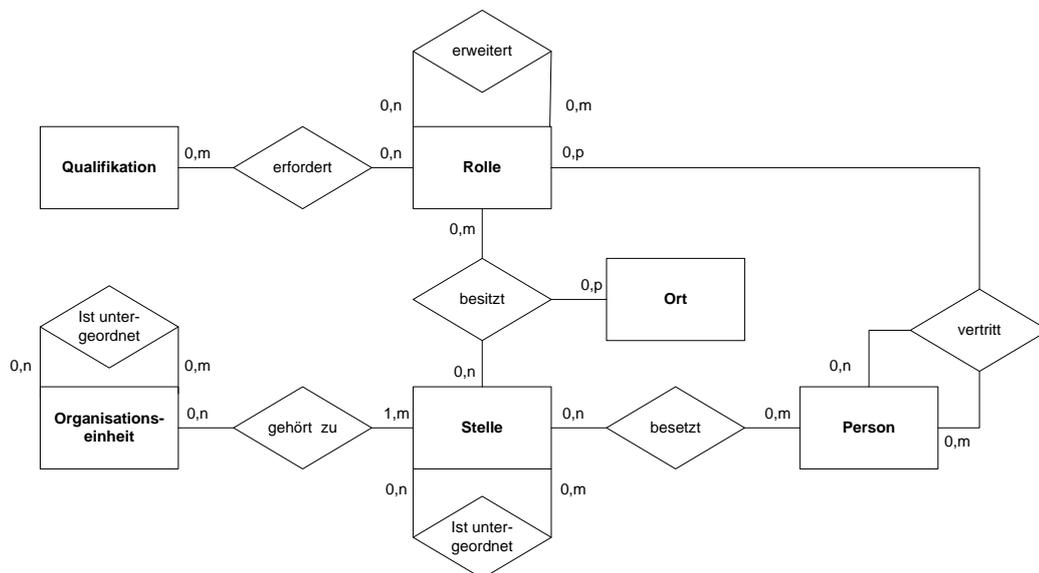


Abb. 21: Verwendetes Organisations-Metamodell

Es existieren Strukturbeziehungen bei Organisationseinheiten und Stellen, so dass Hierarchien abgebildet werden können. Ähnliches gilt für Rollen, welche andere Rollen aggregieren oder erweitern können. Eine Rolle fasst alle Qualifikationen zusammen, die zur Ausführung einer Aktivität notwendig sind - dies impliziert, dass für jede Aktivität eine Rolle existieren muss. Rollen sind einerseits mit Stellen, andererseits mit einem Ausführungsort verknüpft. Dadurch ist es möglich festzuhalten, wo ein Mitarbeiter eine Aktivität ausführt. Weiterhin existiert ein Beziehungstyp „vertritt“, wodurch festgehalten werden kann, wel-

cher Mitarbeiter einen anderen Mitarbeiter in einer bestimmten Rolle vertritt. Wäre die Kante zwischen „vertritt“ und „Rolle“ nicht vorhanden, würde dies zu einer zu groben Vorstellung führen, da man davon ausginge, dass der Vertreter alle Rollen des zu vertretenden Mitarbeiters übernehmen würde.

Es ist vorgesehen, dass alle Entitäten im Organisations-Metamodell über Attribute verfügen. So haben alle Entitäten das Attribut „Name“, über welches sie identifiziert werden können. Wenn also von der Rolle „Entwickler“ gesprochen wird, ist die Rolle mit den Namen „Entwickler“ gemeint.

Hierbei muss beachtet werden, dass die Namen der Entität „Person“, dem Initiator in den Verlaufsdaten entsprechen müssen, da andernfalls keine Übereinstimmung festgestellt werden kann - ein Mining wäre damit nicht möglich.

5.4 Entwicklung der Mining-Methode

5.4.1 Anforderungen an die Verlaufsdaten

Um Organisationswissen in den Mining-Prozess einbeziehen zu können ist es notwendig, Verlaufsdaten und Organisationswissen miteinander zu verknüpfen. Es muss daher der Bearbeiter einer jeden Aktivität bekannt sein, sodass eine Abbildung auf das Organisationsmodell erfolgen kann. Wie bereits im vorherigen Kapitel erläutert, ist es dabei essentiell, dass die Namen der Bearbeiter im Log den Namen der Personen im Organisationsmodell entsprechen.

Darüber hinaus ist eine Aktivität typischerweise durch ein Start- und ein Endereignis begrenzt. Es ist notwendig, dass beiden Ereignissen der gleiche Bearbeiter zugeordnet wird, damit jeder Aktivität eindeutig genau ein Bearbeiter zugeordnet werden kann.

Alternativ wäre auch eine atomare Betrachtung von Aktivitäten denkbar, wodurch jede Aktivität nur einmal in den Verlaufsdaten aufgeführt wird (typischerweise mit dem Endzeitpunkt als Zeitstempel). Ein Vorteil dieser Alternative liegt darin, dass die Umsetzung der in Kapitel 5.3.1 vorgestellten Ansätze mit bisherigen Ansätzen kombiniert werden könnte, welche meist mit dieser Form der Verlaufsdaten arbeiten.

Bei der Ausführung von Prozessen existieren häufig auch Aktivitäten, welche IT-technisch nicht unterstützt und deshalb manuell ausgeführt werden. Die auf diese Weise entstandenen, aber elektronisch nicht protokollierten Verlaufsdaten müssen folglich auf anderen Wegen beschafft werden. Die meisten dieser Aktivitäten werden in verschiedenen Dokumenten, Formularen oder Akten dokumentiert, sodass eine Idee darin liegen könnte, diese Dokumente für die

Gewinnung der notwendigen Informationen zu verwenden. Darauf basierend stellen Wolf und Rosenblum⁹⁶ einen Ansatz vor, welcher sich mit der Akquisition von Verlaufsdaten über einen nicht automatisierbaren Prozess für eine Fallstudie beschäftigt. Diese oder ähnliche Vorgehensweise könnten für die vorliegende Problemstellung herangezogen werden⁹⁷.

5.4.2 Auswahl des Zielmodells

Als Zielmodell kommen aufgrund der häufigen Verwendung entsprechend des State-of-the-Art (siehe Kapitel 3.9) insbesondere drei Modelltypen in Betracht – der Control-Flow Graph, das Petri-Netz und die (e)EPK.

Gegen CFGs und Petri-Netze spricht die relativ schwierige Lesbarkeit für Personen ohne besondere Fachkenntnisse, wohingegen die EPK intuitiv verständlicher erscheint. Weiterhin bietet die eEPK Konzepte für die Modellierung von Organisationseinheiten, welche bei Petri-Netzen und CFGs nicht zur Verfügung stehen. Gerade im vorliegenden Kontext, in welchem Organisationswissen eine sehr hohe Relevanz darstellt, ist diese Tatsache von besonderer Bedeutung.

Würde die Umsetzung der vorgeschlagenen Ansätze in Kombination mit bereits vorhandenen Ansätzen erfolgen, wäre jedoch auch die Verwendung von Petri-Netzen denkbar. Nahezu alle Arbeiten von van der Aalst, welcher als Pionier auf dem Gebiet des Process Mining anzusehen ist, verwenden diesen Modelltyp, sodass eine Kombination mit dessen verhältnismäßig weit entwickelten Ansätzen sinnvoll erscheint.

Es ist also festzuhalten, dass die Auswahl des Zielmodells auf der Auswahl der umzusetzenden Ansätze basieren muss – spielt beispielsweise die Annotation von Organisationseinheiten eine Rolle, so würde man eine eEPK generieren, während im Falle der Kombination mit bestehenden Ansätzen am ehesten das Petri-Netz gewählt würde.

⁹⁶ Wolf, A.L., Rosenblum, D.S.: A Study in Software Process Data Capture and Analysis, In: Proceedings of the Second International Conference on the Software Process (1993), URL: <http://users.ece.utexas.edu/~perry/education/SE-Intro/wolf-rosenblum.pdf>, Abrufdatum: 15.08.2011, S. 119ff.

⁹⁷ In Anlehnung an: Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, S. 67, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.

5.4.3 Formalisierung der Ansätze

5.4.3.1 Bemerkung

Im Folgenden wird davon ausgegangen, dass die Kopplung von Verlaufsdaten und Organisationsmodell, wie in den vorherigen Kapiteln beschrieben, durchgeführt wurde. Es kann also von jedem Initiator innerhalb der Verlaufsdaten auf die ihm zugeordneten Rollen geschlossen werden. Darauf basierend werden nun diejenigen Ansätze formalisiert, welche für die Bearbeitung der bekannten Probleme im Process Mining relevant sind. Es handelt sich dabei um die Erkennung und Behandlung von doppelten Tasks und Rauschdaten, sowie die Erkennung von Parallelität.

Die vorgestellten Potentiale, welche sich mit Bearbeiterzuordnungsregeln und dem Prozessablauf in der Organisation beschäftigen, werden hierbei nicht weiter berücksichtigt, da diese den Mining-Prozess nicht unterstützen, sondern lediglich Zusatzinformationen liefern. Die vorgeschlagene Prozessidentifikation- und Separierung könnte im Rahmen einer Post-Processing Phase relevant werden, um das Zielmodell durch das Clustering übersichtlicher zu machen, was jedoch ebenfalls keine Auswirkungen auf den eigentlichen Mining-Prozess hat.

5.4.3.2 Behandlung doppelter Tasks

Sei A die Menge der im Log vorkommenden Aktivitäten, M die Menge der im Unternehmen beschäftigten Mitarbeiter (welche aus dem Organisationsmodell hervorgehen) und $M_A(a)$, mit $a \in A$ die Menge der Mitarbeiter, welche die Aktivität a innerhalb der Verlaufsdaten mindestens einmal ausführen. Mit dem Startwert $A = \{\}$ wird A mittels einer Iteration über die Verlaufsdaten wie folgt berechnet (1).

```
A = {}
FOREACH entry FROM log {
  A = A UNION {entry.activity}
}
```

Es wird für jedes $a \in A$ eine Menge $M_A(a) = \{\}$ erzeugt, welche anschließend durch erneute Iteration über die Verlaufsdaten gefüllt wird. Da hierdurch eine Zuordnung von Aktivitäten zu Mitarbeitern erfolgt ist (2), kann für jede Aktivität die minimale Menge $R_{min}(a)$, mit $a \in A$ der dafür notwendigen Rollen ermittelt werden, indem die Schnittmenge der Rollen der in $M_A(a)$ befindlichen Mitarbeiter berechnet wird (3). Falls die Schnittmenge der Rollen eines Mitarbeiters mit der bisherigen Schnittmenge leer ist, wird ein weiteres $R_{min}(a)$ für die betreffende Aktivität erzeugt. Auf diese Weise können für alle

Aktivitäten mehrere minimale Rollenmengen entstehen – die Anzahl dieser Mengen entspricht der Anzahl der tatsächlichen Aktivitäten, welche durch a repräsentiert werden. Ein Beispielcode könnte wie folgt aussehen:

```

FOREACH activity FROM A {
    mA[activity] = {}
}

FOREACH entry FROM log {
    mA[entry.activity] = mA[entry.activity] UNION
                        {entry.initiator}
}

FOREACH activity FROM A {
    FOREACH initiator FROM mA[activity] {
        validIntersectionFound = false
        FOR (i=0, i<=maxKey(rMin[activity], i++)) {
            IF rMin[activity][i] INTERSECT initiator.roles != {}
                THEN validIntersectionFound = true
                    rMin[activity][i] = rMin[activity][i] INTERSECT
                                initiator.roles
        }
        IF validIntersectionFound = false
            THEN rMin[activity][i] = initiator.roles
    }
}

```

Angenommen es wurde ein doppelter Task identifiziert, es wurden also beispielsweise für die Aktivität $a = B$ mehrere minimale Schnittmengen gefunden, dann muss eine Umbenennung der Aktivität B in B1, B2, usw. erfolgen. Die Entscheidung, um welches B es sich handelt, wird aufgrund der Rollenzugehörigkeit des Mitarbeiters getroffen. Wenn also im Log die Aktivität B von M1 ausgeführt wird, welcher über alle Rollen der zweiten minimalen Rollenmenge verfügt, dann wird B in B2 umbenannt.

Es könnte der Fall eintreten, dass M1 beide minimalen Rollenmengen erfüllt, sodass nicht entschieden werden kann, auf welche Aktivität abgebildet werden soll. Da es jedoch unstrittig ist, dass es sich um einen doppelten Task handelt, muss die konkrete Prozessinstanz aus dem Log entfernt werden, da andernfalls die Korrektheit der Verlaufsdaten nicht garantiert werden kann (4).

```

FOREACH entry FROM log {
    IF countMinRoleSetsOf(entry.activity) > 0
        THEN
            IF hasUniqueMinRoleSet(entry.initiator, entry.activity)
                THEN entry.activity = entry.activity +
                    getIndexOfMinRoleSet(entry.initiator, entry.activity)
            ELSE log.deleteInstance(entry.caseID)
}

```

Zusammenfassend lässt sich die Erkennung und Behandlung doppelter Tasks in 4 Schritten abwickeln:

1. Identifikation der im Log enthaltenen Aktivitäten
2. Zuordnung von Mitarbeitern zu Aktivitäten
3. Berechnung minimaler Rollenschnittmengen
4. Bereinigung des Verlaufsdaten

5.4.3.3 Erkennung von Rauschdaten

Bei der Erkennung doppelter Tasks kann es, wie in Kapitel 5.3.1.3 beschrieben, dazu kommen, dass minimale Rollenschnittmengen erkannt werden, welche verhältnismäßig selten für die Ausführung einer Aktivität relevant sind. Ein solches Ergebnis ist insbesondere dann zu erwarten, wenn es sich entweder um einen Fehler in den Verlaufsdaten oder um einen Fehler im Organisationsmodell, also um Rauschdaten handelt.

Die Unterscheidung von doppelten Tasks und Rauschdaten soll hier über einen Parameter q (Quote) gesteuert werden, welcher eine prozentuale Schwelle darstellt, ab wann eine minimale Rollenschnittmenge als Rauschen zu interpretieren ist. Um dies zu erreichen muss zunächst ermittelt werden, wie oft jede Aktivität in den Verlaufsdaten enthalten ist. Die Funktion $count : log \times activity \rightarrow \mathbb{N}$ ermittelt diese Werte durch Iteration über das Log.

Weiterhin muss berechnet werden, wie oft eine minimale Rollenschnittmenge für die Ausführung einer Aktivität relevant ist. Hierbei muss allerdings berücksichtigt werden, dass einige Fälle aufgrund fehlender Eindeutigkeit nicht beachtet werden dürfen (siehe hierzu Kapitel 5.4.3.2). Dementsprechend muss der letzte Schritt des vorherigen Kapitels wie folgt angepasst werden:

```
// Entfernen der nicht-eindeutigen Aktivitäten (Instanzen)
FOREACH entry FROM log {
    IF countMinRoleSetsOf(entry.activity) > 0 AND
       NOT hasUniqueMinRoleSet(entry.initiator, entry.activity)
    THEN log.deleteInstance(entry.caseID)
}

// Zählen der Aktivitätsvorkommnisse nach erster Bereinigung
FOREACH activity FROM A {
    incidents[activity] = count(log, activity)
}

// Zählen der Vorkommnisse der minimalen Rollenschnittmengen
FOREACH rMin[activity][i] from A, [0;maxKey(rMin[activity])] {
    incidents[activity][i] = countIncid(log, rMin[activity][i]);
}
```

```

// Bereinigung der Verlaufsdaten
FOREACH entry FROM log {
    i = getIndexOfMinRoleSet(entry.initiator, entry.activity)
    IF incidents[entry.activity][i] /
        incidents[entry.activity] < p
    THEN log.deleteInstance(entry.caseID)
    ELSE entry.activity = entry.activity + i
}

```

Instanzen, in denen Rauschen identifiziert wurde, werden aus dem Log entfernt.

5.4.3.4 Parallelität

Zusammenfassend lassen sich aus Kapitel 5.3.1.2 zwei Potentiale des Organisationswissens ableiten, um Parallelität zu erkennen bzw. zu widerlegen:

1. Bei atomarer Betrachtung von Aktivitäten: Eine potentielle Parallelität kann ausgeschlossen werden, falls die betroffenen Aktivitäten immer von derselben Person ausgeführt werden. Hierzu ist ausschließlich die Initiator-Information notwendig.
2. Bei nicht-atomarer Betrachtung von Aktivitäten: Echte Parallelität kann ausgeschlossen werden, falls die betroffenen Aktivitäten immer von derselben Person ausgeführt und an unterschiedlichen Orten ausgeführt werden. Hierzu ist die Kenntnis über die (genau eine) notwendige Rolle, sowie den Ausführungsort notwendig.

Ein Problem besteht im zweiten Ansatz, da eine Voraussetzung in der Kenntnis der korrekten Rolle liegt. Zwar lässt sich durch die bisher beschriebenen Ansätze eine minimale Rollenschnittmenge für jede Aktivität ermitteln, jedoch ist eine Mächtigkeit dieser Menge von 1 nicht garantiert. Eine Lösung könnte in der Gleichbenennung von Aktivitäten und den zugehörigen Rollen liegen - da dies jedoch im Allgemeinen nicht gegeben und das Problem mit den aktuellen Ansätzen nicht zu lösen ist, wird im Folgenden nur die Variante mit atomarer Betrachtung von Aktivitäten formalisiert.

Ausgangspunkt hierfür ist, dass bereits ein Prozessmodell abgeleitet wurde – die Stellen, an denen potentiell Parallelität vorhanden ist, sind dementsprechend mit AND-Konnektoren oder einem äquivalenten Konstrukt gekennzeichnet. A_1, \dots, A_n seien die Mengen der in den parallel verlaufenden Pfaden p_1, \dots, p_n mit $n \geq 2$ enthaltenen Aktivitäten. I_1, \dots, I_m seien die Mengen der Initiatoren, die die Aktivitäten in der Menge $A_1 \cup \dots \cup A_n$ in allen Prozessinstanzen c_1, \dots, c_m ausführen.

Wenn $\prod_{i=1}^m |I_i| = 1$ gilt, bedeutet das, dass innerhalb jeder Prozessinstanz die potentiell parallel ablaufenden Aktivitäten von derselben Person ausgeführt werden, wodurch die Parallelität widerlegt ist.

Ist das der Fall, muss das Prozessmodell entsprechend korrigiert werden. Statt des ursprünglichen AND-Konstrukts mit parallelen Pfaden wird nun ein XOR-Konstrukt mit allen möglichen Abfolge-Kombinationen als Pfade benötigt. Diese Abfolge-Kombinationen berechnen sich wie folgt:

Zunächst sollte festgehalten werden, dass es $n!$ Möglichkeiten gibt, die ursprünglichen Pfade neu anzuordnen (in der Statistik handelt es sich hierbei um eine geordnete Stichprobe ohne Wiederholung) – es wird also $n!$ XOR-Pfade geben. Demzufolge seien $q_1, \dots, q_n!$ mit $q_i \neq q_j \forall i, j \in 1, \dots, n!$ die neuen Pfade (Abfolge von Aktivitäten), welche mit einem umklammernden XOR die ursprüngliche Variante ersetzen.

5.4.4 Umsetzung der vorgeschlagenen Potentiale

Die in Kapitel 5.3.1 erarbeiteten Potentiale sind aufgrund Ihrer Zielsetzungen in unterschiedlichen Phasen umzusetzen. Die Erkennung und Behandlung von Rauschen und doppelten Tasks ist beispielsweise am ehesten in die Pre-Processing Phase einzuordnen, da dort eine Bereinigung bzw. Aufbereitung der Verlaufsdaten erfolgt.

Die Behandlung von Parallelität ist hingegen der Processing Phase zuzuordnen, da hier eine Unterstützung des eigentlichen Mining Algorithmus gegeben ist. Allerdings reichen die vorgestellten Potentiale nicht aus, um damit ein komplettes Processing zu realisieren, weshalb immer auf die klassischen Ansätze als Basis zurückgegriffen werden muss. Denkbar wäre deshalb eine Kombination des vorgestellten Ansatzes zur Parallelitätserkennung mit bereits etablierten Ansätzen, was konkret über die Integration des Vorgehens in ein existierendes Processing gestaltet werden könnte.

Die Prozessidentifikation und –separierung, ebenso wie die Annotation von Organisationswissen am Prozessmodell sind im Post Processing zu sehen. Während es sich bei Ersterem um ein Clustering handelt, welches die Lesbarkeit des Zielmodells erhöht, werden bei der Annotation von Organisationswissen zusätzliche Informationen generiert. Beide Aspekte haben jedoch keine Auswirkung auf die Ableitung der Prozesslogik. Hierbei ist zu beachten, dass vorhandene Ansätze unterschiedliche Output-Modelle produzieren. Vor der Annotation von Organisationswissen ist es deshalb je nach Modelltyp erforderlich eine Transformation nach EPK durchzuführen.

Das Ableiten von Bearbeiterzuordnungsregeln ist in den Ansatz nicht einzubeziehen, da es sich um einen losgelösten Mining-Aspekt handelt, welcher keine Auswirkung auf die abzuleitende Prozesssicht hat. Es werden hierbei ebenfalls zusätzliche (organisatorische) Informationen erzeugt, welche im Rahmen dieser Arbeit nicht weiter verfolgt werden. Prinzipiell ist jedoch ein Vorgehen ähnlich dem Staff Assignment Mining denkbar.

6 Prototyping

6.1 Einführung

Um die entwickelte Mining-Methode mit Beispieldaten evaluieren zu können, soll im Rahmen der Arbeit eine erste prototypische Implementierung in Form eines ProM-Plugins erfolgen. Es ist für einen Prototypen charakteristisch, dass nur die wesentlichen Funktionen und die Benutzungsoberfläche umgesetzt werden. Es muss deshalb zunächst geklärt werden, welche Funktionen als „wesentlich“ zu betrachten und damit umzusetzen sind.

Da es sich um eine Methode handelt, welche Organisationswissen in besonderer Weise berücksichtigt, ist es essentiell, dass Organisationswissen im Prototyp eingebunden und verwendet werden kann. Gleiches gilt für Verlaufsdaten, welche ebenso die Basis für das Mining bilden. Es muss also definiert werden, in welcher Form bzw. in welchem Format diese Daten repräsentiert werden, um anschließend verwendet werden zu können.

Weiterhin soll eine erste Evaluation erfolgen, sodass festgelegt werden muss, welche der in Kapitel 5.4.3 formalisierten Ansätze implementiert werden.

6.2 ProM

Bei ProM handelt es sich um ein Open-Source Framework welches zur Implementierung von Ansätzen im Umfeld des Process Mining entwickelt wurde. Dabei kann es sich neben konkreten Mining-Algorithmen ebenfalls um Analyse-, Monitoring- und ähnliche Ansätze handeln. Initiiert wurde das Projekt von der Eindhoven University of Technology, welcher in diesem Umfeld eine Vorreiterrolle zukommt.

ProM hat sich in den letzten Jahren zum Standard Process Mining Framework im akademischen Umfeld etabliert, wodurch die aktuelle Version 6 mittlerweile 88 offizielle Plugins zählt, zuzüglich einer unbekanntenen Anzahl inoffizieller Plugins. Weiterführende Informationen sind unter <http://www.processmining.org> frei zugänglich.

6.3 Repräsentation der Verlaufsdaten

Da es sich bei ProM um ein Process Mining Framework handelt, stehen bereits diverse Möglichkeiten für den Import von Verlaufsdaten zu Verfügung. ProM-Import extrahiert beispielsweise MXML-formatierte Logdaten aus einer Viel-

zahl von Informationssystemen⁹⁸. Das Standardformat für Verlaufsdaten in ProM nennt sich XES (= „Extensible Event Stream“) und ist wie folgt aufgebaut.

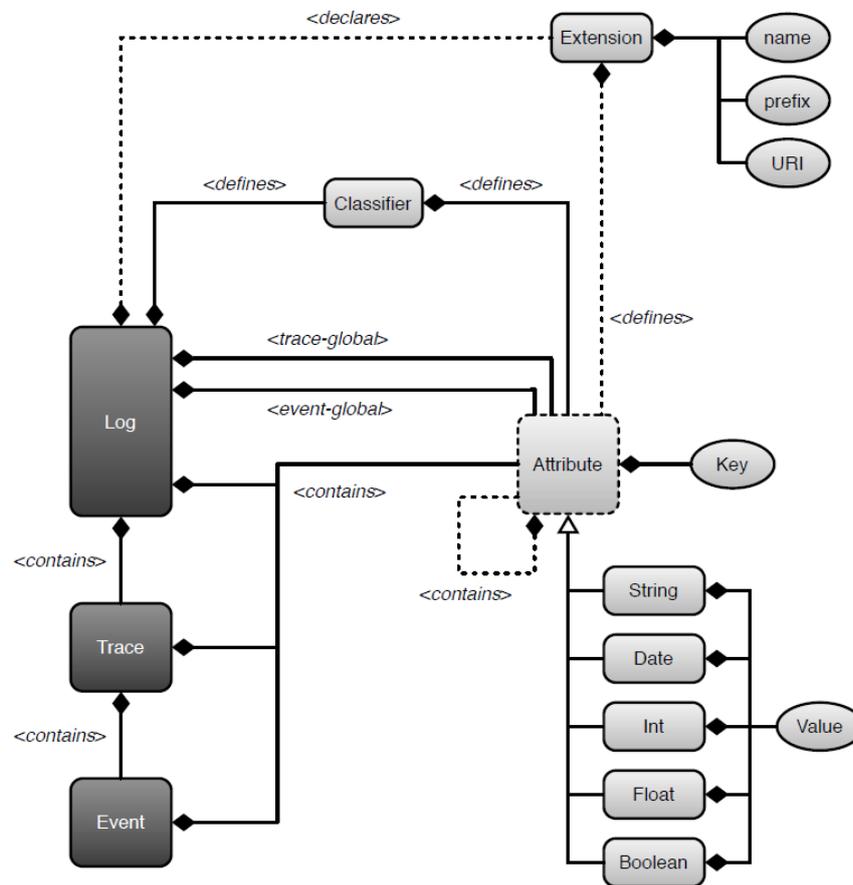


Abb. 22: XES 1.0 Standard Definition⁹⁹

Ein Log besteht demnach aus mehreren Traces (Cases), welche wiederum aus mehreren Events (Ereignissen) bestehen. Die meisten Ansätze arbeiten mit Verlaufsdaten, welche ausschließlich die Endereignisse von Aktivitäten beinhalten. Die Darstellung von Logs mit Start- und Endereignissen ist jedoch problemlos realisierbar. Bei XES handelt es sich um ein spezielles XML-Format.

⁹⁸ ProMimport – URL: <http://www.processmining.org/promimport/start>, Abrufdatum: 27.08.2011.

⁹⁹ Günther, C.W. (Fluxicon Process Laboratories): XES – Standard Definition (Draft) Version 1, November 2009, S. 2.

6.4 Repräsentation des Organisationsmodells

Da die Verwendung von Organisationswissen im Process Mining neu ist, verfügt ProM bisher nicht über entsprechende Implementierungen, mit welchen Organisationsmodelle importiert werden könnten. Daher ist es notwendig, neben den Processing-Klassen, ebenfalls eine Import-Funktion zu realisieren.

Da ein Organisationsmodell, wie in Kapitel 5.3.3 beschrieben, am besten durch eine Tabellenstruktur dargestellt werden kann, soll ebenso wie bei den Verlaufsdaten eine XML-Struktur verwendet werden. Da es sich um einen Prototypen handelt, werden zunächst nur die wesentlichen Entitäten Person, Stelle, Rolle und Organisationseinheit umgesetzt. Beispielhaft sieht eine XSOM-Datei (XML Simple Organizational Model) wie folgt aus.

```
<?xml version="1.0" encoding="UTF-8"?>
<OrganizationalModel>

  <persons>
    <person name="Peter">
      <jobs>
        <job name="Service-Ingenieur" />
      </jobs>
    </person>
    <person name="Joe">
      <jobs>
        <job name="Reklamations-Ingenieur" />
      </jobs>
      <representations>
        <representation person="Peter" role="C" />
      </representations>
    </person>
  </persons>

  <jobs>
    <job name="Reklamations-Ingenieur">
      <roles>
        <role name="E" />
      </roles>
      <OrganizationalUnits>
        <OrganizationalUnit name="Auftragsabwicklung" />
      </OrganizationalUnits>
    </job>
    <job name="Service-Ingenieur">
      <roles>
        <role name="C" />
      </roles>
      <OrganizationalUnits>
        <OrganizationalUnit name="Auftragsabwicklung" />
      </OrganizationalUnits>
    </job>
  </jobs>
</OrganizationalModel>
```

```

<OrganizationalUnits>
  <OrganizationalUnit name="Service">
  </OrganizationalUnit>
  <OrganizationalUnit name="Auftragsabwicklung">
    <subordinations>
      <subordination name="Service" />
    </subordinations>
  </OrganizationalUnit>
</OrganizationalUnits>

<roles>
  <role name="A" />
  <role name="B" />
  <role name="C" />
  <role name="D" />
  <role name="E" />
</roles>
</OrganizationalModel>

```

Alle Entitäten haben ein Attribut „Name“ über welches Sie identifiziert werden. Eine Person enthält weiterhin eine Menge von Jobs (Stellen), ein Job wiederum beinhaltet eine Menge von Rollen, sowie eine Menge von Organisationseinheiten. Im Falle der Organisationseinheiten wird außerdem bereits das Strukturkonzept umgesetzt, was in gleicher Weise auf die übrigen Entitäten übertragen werden kann. Bei Personen kann ferner die Beziehung „vertritt“ durch Knoten mit dem Namen „representation“ realisiert werden. Eine konkrete Person vertritt also die Personen, welche im Attribut „person“ hinterlegt ist, in der Rolle, welche im Attribut „rolle“ hinterlegt ist.

Die Komplettierung dieser Struktur im Sinne des vorgestellten Organisations-Metamodells kann analog vorgenommen werden.

6.5 Auswahl der zu implementierenden Ansätze

Um eine Kombination der Mining Methode mit bereits vorhandenen Ansätzen (bzw. Plugins in ProM) umsetzen zu können, soll für den Prototyp die Erkennung und Behandlung von doppelten Tasks und Rauschdaten in Form eines Pre-Processing implementiert werden. Für dieses Pre-Processing sollen Verlaufsdaten und Organisationsmodell als Input und die aus dem Algorithmus resultierenden überarbeiteten Verlaufsdaten als Output dienen. Zusätzlich soll der Output einen Report generieren, welcher die am Log durchgeführten Änderungen zusammenfasst.

Die Output-Verlaufsdaten können dann als Input für ein beliebiges, bereits vorhandenes ProM-Plugin verwendet werden, wodurch eine große Anzahl an Kombinationen mit anderen Ansätzen ermöglicht wird.

6.6 Implementierung

Das ProM-Framework selbst wurde in Java entwickelt, sodass auch die zugehörigen Plugins in Java realisiert werden sollten. Um den Prototypen wie beschrieben realisieren zu können, bedarf es der Implementierung von zwei Plugins: ein Import-Plugin, welches die Aufnahme des Organisationsmodells ermöglicht und ein Pre-Processing Plugin, welches die eigentlichen Funktionalitäten zur Verfügung stellt.

Das Import-Plugin nimmt die vom Benutzer ausgewählte xsom-Datei auf und erzeugt daraus ein Objekt der Klasse SimpleOrganizationalModel, welche das Organisationsmodell repräsentiert. Dieses Objekt dient als Ausgangspunkt für alle weiteren Aktionen, die mit dem Organisationsmodell in Verbindung stehen. Für jede Entität aus dem Organisationsmetamodell existiert darüber hinaus eine eigene Klasse, welche spezifische Methoden anbietet - beispielsweise kann über ein Objekt der Klasse „Person“ geprüft werden, ob die aktuelle Person eine andere Person in einer bestimmten Rolle vertritt. Eine ausführliche Dokumentation der implementierten Klassen und Funktionen liegt dem Quellcode als JavaDoc bei. Ein erfolgreicher Import des Organisationsmodells sieht dann wie folgt aus.

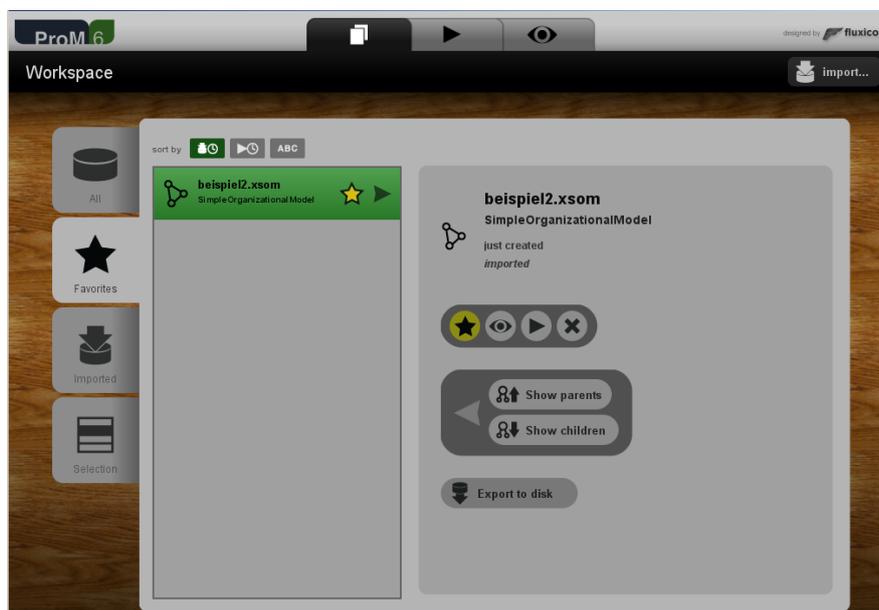


Abb. 23: Prototyp: Import Organisationsmodell

Die Implementierung eines Import-Plugins für die Verlaufsdaten ist nicht notwendig, da dieses bereits existiert. Allerdings werden durch die Verwendung von Organisationswissen weitere Funktionalitäten notwendig, welche durch die

bisherigen Implementierungen nicht abgedeckt werden. Es wurde deshalb eine Helper-Klasse entwickelt, welche diese Funktionslücken schließt.

Da das Pre-Processing, neben den bereinigten Verlaufsdaten, einen Report im HTML-Format für den Output generieren werden soll, wurde die Klasse HTMLResultBuilder implementiert, welche die dafür notwendigen Methoden bereitstellt. Hierbei können die Verlaufsdaten analysiert und für den Report aufbereitet werden. Dies ist insbesondere im Hinblick auf die Nachvollziehbarkeit der Pre-Processing Aktionen wichtig und bewirkt gleichzeitig eine ideale Kapselung der unterschiedlichen Funktionalitäten.

Da hiermit alle notwendigen Voraussetzungen geschaffen wurden, kann mit der Umsetzung des Pre-Processing Plugins begonnen werden. Um auch hier eine übersichtliche Kapselung der einzelnen Aktionen zu erreichen, werden zunächst die Verlaufsdaten gemäß dem vorgestellten Ansatz auf Rauschen untersucht und bereinigt, anschließend werden die nicht-eindeutigen Aktivitäten (bzw. die entsprechenden Cases) entfernt und dann die doppelten Tasks umbenannt (siehe Kapitel 5.4.3.2).

An dieser Stelle ist es wichtig darauf hinzuweisen, dass die Reihenfolge der Aktionen keinesfalls willkürlich ist. Die Entfernung der Rauschdaten muss zwingend als erstes durchgeführt werden, da dadurch potentielle doppelte Tasks entfernt werden. Anschließend müssen die nicht-eindeutigen Aktivitäten entfernt werden, da diese nicht für die Erkennung doppelter Tasks herangezogen werden können. Es könnte in diesen Fällen nicht entschieden werden, um welche Aktivität es sich handelt. Erst nach Abschluss dieser Vorselektionen kann die Behandlung doppelter Tasks erfolgen.

Der Pre-Processing Report sieht nach dem Vorgang exemplarisch wie folgt aus.

ProM 6 designed by fluxicon

Result Summary Create new...

Organizational Preprocessing Report

Noise-Parameter: 30 %

Source-Log

Cases: 7
 # Tasks: 5
 # Tasks (Executions): 23

| Task Overview | Task | Executions | #Initiators | Initiators |
|---------------|------|------------|-------------|---------------------|
| | A: | 7 | 1 | Peter |
| | E: | 5 | 3 | Tom, Giovanna, Honk |
| | D: | 7 | 1 | Giovanna |
| | C: | 2 | 1 | Joe |
| | B: | 2 | 1 | Peter |

Execute Preprocessing

| Action | Reason |
|---|-------------------|
| Delete all Cases which include Task E executed by Honk. | Undetermined Task |
| Rename all Tasks E executed by Tom to E0. | Duplicated Task |
| Rename all Tasks E executed by Giovanna to E1. | Duplicated Task |

Cleaned Log

Cases: 6
 # Tasks: 6
 # Tasks (Executions): 20

| Task Overview | Task | Executions | #Initiators | Initiators |
|---------------|------|------------|-------------|------------|
| | A: | 6 | 1 | Peter |
| | D: | 6 | 1 | Giovanna |
| | E0: | 3 | 1 | Tom |
| | E1: | 1 | 1 | Giovanna |
| | C: | 2 | 1 | Joe |
| | B: | 2 | 1 | Peter |

Abb. 24: Prototyp: Organizational Pre-Processing Report

7 Evaluation

7.1 Einführung und Vorgehensweise

In diesem Kapitel soll die zuvor entwickelte Methode mit Echtdateien evaluiert werden. Es wurde ein Partner gesucht, welcher die erforderlichen Logdateien in elektronischer Form, ebenso wie Organisationswissen zur Verfügung stellen kann. Dabei war es einerseits wichtig, dass der zu untersuchende Geschäftsprozess nicht zu komplex und andererseits von der Anzahl der Aktivitäten her übersichtlich ist, damit eine Aussage darüber getroffen werden kann, wie nah das Zielmodell an den tatsächlichen Prozess reicht.

Auf den genannten Daten soll ein Process Mining mit bekannten Ansätzen durchgeführt werden, wobei jeweils eine Variante das hier entwickelte Pre-Processing beinhaltet, die andere nicht. Da die Evaluation über das Process Mining Werkzeug ProM erfolgen soll, beschränkt sich die Auswahl der bekannten Ansätze auf die darin implementierten.

Konkret wurden für die Evaluation der Alpha-Algorithmus und der Genetic Miner (beide aus dem Bereich der algorithmischen Methoden) ausgewählt. Der Alpha-Algorithmus dient in einer Vielzahl von Arbeiten zum Process Mining als Referenz, sodass dieser Benchmark auch im vorliegenden Kontext sinnvoll erscheint. Beim Genetic Process Mining sprechen viele Autoren heute von einer eigenen Methoden-Gattung neben den algorithmischen Ansätzen, was mitunter auf die steigende Relevanz zurückzuführen ist. Aus diesem Grund soll auch hier evaluiert werden, ob die entwickelte Methode diesen Mining-Prozess unterstützen kann.

7.2 Leginda

7.2.1 Vorstellung

Die Leginda GmbH ist eine Kooperation des Softwarehauses META-LEVEL Software AG mit dem Übersetzungsdienstleister Lector GmbH in Saarbrücken. Es handelt sich um eine Online-Übersetzungsplattform, wobei META-LEVEL für die technische und Lector für die fachliche Seite zuständig ist.

Leginda startete am 1. Januar 2008 als erste Plattform dieser Art in Deutschland und wurde im März desselben Jahres mit dem Innovationspreis 2008 in der Kategorie Branchensoftware ausgezeichnet.

Nach dem Konzept der Leginda GmbH sind an einem Übersetzungsfall mehrere Akteure beteiligt – ein Kunde, ein Controller und ein Übersetzer. Die Platt-

form wurde demnach als Workflow-System realisiert, welches die Kernfunktionalitäten mitprotokolliert und in einer Datenbank ablegt.

7.2.2 Prozessbeschreibung

Ein Übersetzungsfall (Case) wird gestartet (*create_translation_case*), indem ein Kunde ein zu übersetzendes Dokument in das System lädt. Das System analysiert dieses Dokument und erstellt auf Basis der Ergebnisse automatisch ein Angebot, welches dem Kunden angezeigt wird. Wenn das Dokument nicht kalkulierbar oder der Kunde mit dem Angebot nicht zufrieden ist wird eine manuelle Kalkulation des Controllers angefordert (*request_calculation*). Der Kunde hat weiterhin die Möglichkeit das ihm unterbreitete Angebot abzulehnen und in Folge daraus den Geschäftsvorfall abubrechen (*cancel_case*). Im ersten Fall erstellt der Controller ein neues Angebot und übermittelt dieses dem Kunden (*offer_calculation*). Der Kunde hat nun wieder die Möglichkeit ein weiteres Angebot anzufordern oder das Angebot anzunehmen (*order_translation*). Wurde die Übersetzung in Auftrag gegeben, führt der Controller eine Selektion von Übersetzern durch, und bietet diesen den Geschäftsvorfall zu einem zuvor (automatisch oder manuell) kalkulierten Entgelt an (*offer_translation*). Solange kein Übersetzer den Auftrag angenommen hat, kann der Controller das Angebot jederzeit zurückziehen (*withdraw_translation*). Nimmt kein Übersetzer den Auftrag an (*reject_offer*), wird das Angebot überarbeitet und erneut angeboten, andernfalls (*accept_offer*) wird der Übersetzer den Auftrag bearbeiten und ein Übersetzungsdokument ins System laden (*do_translation*). Der Controller prüft nun die Übersetzung und liefert diese an den Kunden (*deliver_translation*), bzw. verlangt Nachbesserung vom Übersetzer (*reject_translation*). In bestimmten Fällen kann der Controller die Lieferung zurückziehen (*withdraw_translation*). Automatisch wird bei der Lieferung die Kundenrechnung generiert und dem Geschäftsvorfall angehängt. Der Kunde kann die Lieferung nach eigener Prüfung akzeptieren (*accept_delivery*) oder zur Nachbesserung zurück an den Controller leiten (*reject_translation*), welcher die Mängel mit Hilfe eines Übersetzers behebt. Nach Auftragsabwicklung kann der Kunde die Abwicklung beurteilen (*rate_case*). Der Controller hat weiterhin zu jedem Zeitpunkt die Möglichkeit den Auftrag abubrechen.

7.3 Vorbereitung der Evaluation

7.3.1 Anonymisierung

Aus datenschutzrechtlichen Gründen dürfen nur anonymisierte Verlaufsdaten aus dem Leginda-System extrahiert und für die Evaluation verwendet werden. Dies hat zur Folge, dass keine personenbezogenen Daten, wie Name, Adresse, Nutzer-ID oder Ähnliches in den Verlaufsdaten enthalten sein dürfen. Aus diesem Grund wurden die Log-Daten so aufbereitet, dass statt der Namen der Initiatoren lediglich deren Stelle, also Controller, Übersetzer oder Kunde festgehalten werden.

Für die durchzuführende Evaluation ist das kein Problem, da im System jede Person nur genau eine Stelle besetzt und keine überlappenden Rollen vorzufinden sind.

Das Organisationsmodell wird demzufolge mit lediglich den drei Personen Controller, Kunde und Übersetzer modelliert, welche jeweils die Stelle Controller, Kunde oder Übersetzer besetzten. Da, wie bereits angemerkt, keine überlappenden Rollen existieren (d.h. es handelt sich um eine 1:1 Beziehung zwischen Rolle und Stelle) wird einem Controller die Rolle „A“, einem Kunden die Rolle „B“ und einem Übersetzer die Rolle „C“ zugeordnet.

An dieser Stelle sei ergänzend angemerkt, dass es keinesfalls ungewöhnlich ist, den Kunden in das Organisationsmodell einzubeziehen. Dies wird gewöhnlich damit argumentiert, dass es sich beim Kunden um einen zentralen Baustein einer jeden Organisation handelt. Gerade im Fall von Leginda trifft das besonders zu, da der Kunde aktiv das Informationssystem bedient und dadurch jeden einzelnen Geschäftsvorfall selbst initiiert.

7.3.2 Separierung der Verlaufsdaten

Ebenso wie fast alle Softwaresysteme unterliegt auch Leginda einem stetigen Entwicklungsprozess innerhalb dessen sich die Software und insbesondere der Workflow ändert. Im laufenden Jahr 2011 wurden insgesamt drei große Versionsupdates eingespielt, deren Inhalt Auswirkungen auf den Workflow hatten. Aus diesem Grund wurden die extrahierten Verlaufsdaten an den Zeitpunkten der Wirksamkeitsaufnahmen geteilt, sodass die einzelnen Versionen getrennt voneinander analysiert werden können. Durch dieses Vorgehen werden eventuelle signifikante Workflowanpassungen transparent und es können Aussagen über die Workflowentwicklung getroffen werden.

Die folgende Tabelle gibt einen Überblick über die extrahierten Verlaufsdaten.

| Version | 2.20 | 2.30 | 2.40 |
|---|-------------|-------------|-------------|
| Wirkbetriebsaufnahme | 20.01.2011 | 12.04.2011 | 16.06.2011 |
| #Cases | 609 | 614 | 732 |
| #Aktivitäten | 3227 | 3358 | 4434 |
| Häufigkeiten der einzelnen Aktivitäten | | | |
| create_translation_case | 607 | 613 | 732 |
| request_calculation | 188 | 184 | 303 |
| cancel_case | 332 | 326 | 300 |
| offer_calculation | 166 | 158 | 285 |
| order_translation | 348 | 345 | 472 |
| offer_translation | 346 | 362 | 525 |
| reject_offer | 39 | 44 | 51 |
| reject_offer_by_all | 1 | 4 | 3 |
| accept_offer | 294 | 295 | 439 |
| do_translation | 286 | 305 | 419 |
| deliver_translation | 290 | 313 | 402 |
| accept_delivery | 289 | 307 | 386 |
| reject_translation | 7 | 17 | 17 |
| withdraw_order | 19 | 67 | 86 |
| withdraw_delivery | 1 | 6 | 1 |
| rate_case | 14 | 12 | 13 |

Tabelle 9: Übersicht - Verlaufsdaten (Evaluation)

In der Evaluation wird demnach der Zeitraum vom 20.01.2011 bis 07.09.2011 mit insgesamt 1.955 Prozessinstanzen, 11.019 Aktivitäten und 16 Aktivitätstypen betrachtet. Die Daten wurden direkt aus der Datenbank des Leginda-Systems im CSV-Format exportiert.

7.3.3 Konvertierung der Verlaufsdaten nach MXML

Die nun separierten und anonymisierten Verlaufsdaten im CSV-Format wurden mit Hilfe des ProM Import Frameworks in das MXML-Format konvertiert, welches wiederum mit XES kompatibel ist. Dieses Format kann anschließend mit ProM weiterverarbeitet werden.

7.4 Organizational Pre-Processing

7.4.1 Vorgehensweise

Das Organizational Pre-Processing wird im Folgenden direkt auf die extrahierten Verlaufsdaten der einzelnen Versionen angewendet. Um die Funktion der Rauschdatenerkennung gesondert sichtbar zu machen, wird dieses Pre-Processing in zwei Varianten durchgeführt – einmal ohne Rauschdatenparameter und einmal mit einem Rauschdatenparameter von 3%.

7.4.2 Version 2.20

| <u>Action</u> | <u>Reason</u> |
|--|-----------------|
| Rename all Tasks create translation case executed by Customer to create translation case0. | Duplicated Task |
| Rename all Tasks create translation case executed by Controller to create translation case1. | Duplicated Task |
| Rename all Tasks order translation executed by Customer to order translation0. | Duplicated Task |
| Rename all Tasks order translation executed by Controller to order translation1. | Duplicated Task |
| Rename all Tasks accept offer executed by Translator to accept offer0. | Duplicated Task |
| Rename all Tasks accept offer executed by Controller to accept offer1. | Duplicated Task |
| Rename all Tasks do translation executed by Controller to do translation0. | Duplicated Task |
| Rename all Tasks do translation executed by Translator to do translation1. | Duplicated Task |
| Rename all Tasks accept delivery executed by Customer to accept delivery0. | Duplicated Task |
| Rename all Tasks accept delivery executed by Controller to accept delivery1. | Duplicated Task |
| Rename all Tasks cancel case executed by Controller to cancel case0. | Duplicated Task |
| Rename all Tasks cancel case executed by Customer to cancel case1. | Duplicated Task |
| Rename all Tasks request calculation executed by Customer to request calculation0. | Duplicated Task |
| Rename all Tasks request calculation executed by Controller to request calculation1. | Duplicated Task |

Tabelle 10: Aktionen des OPP - Version 2.20 - Parameter: 0

Der Pre-Processing Report (Tabelle 10) zeigt, dass die 7 Tasks *create_translation_case*, *order_translation*, *accept_offer*, *do_translation*, *accept_delivery*, *cancel_case* und *request_calculation* als doppelte Tasks erkannt wurden und entsprechend umbenannt wurden.

Eine Verwendung des Rauschdatenparameters mit 3% führt in der Leginda Version 2.20 hingegen zu einem signifikant anderen Ergebnis. Während die Variante ohne Rauschdatenerkennung zu 7 doppelten Tasks führte, wurden hier nur zwei doppelte Tasks erkannt (*do_translation*, *cancel_case*). Einige Prozessinstanzen, welche die übrigen 5 Tasks enthalten, wurden aufgrund des Rauschens aus den Logdaten entfernt, wodurch diese Tasks nicht mehr als doppelt interpretiert wurden (Tabelle 11).

| <u>Action</u> | <u>Reason</u> |
|---|-----------------|
| Delete all Cases which include Task create translation case executed by Controller. | Noise |
| Delete all Cases which include Task order translation executed by Controller. | Noise |
| Delete all Cases which include Task accept offer executed by Controller. | Noise |
| Delete all Cases which include Task accept delivery executed by Controller. | Noise |
| Delete all Cases which include Task request calculation executed by Controller. | Noise |
| Rename all Tasks do translation executed by Controller to do translation0. | Duplicated Task |
| Rename all Tasks do translation executed by Translator to do translation1. | Duplicated Task |
| Rename all Tasks cancel case executed by Controller to cancel case0. | Duplicated Task |
| Rename all Tasks cancel case executed by Customer to cancel case1. | Duplicated Task |

Tabelle 11: Aktionen des OPP - Version 2.20 - Parameter: 3

Eine Rücksprache mit der Leginda GmbH hat ergeben, dass zu Testzwecken eine Entwickler-Rolle im System existiert, welche alle anderen Rollen annehmen kann. Es hat sich weiterhin herausgestellt, dass es im System nicht möglich ist, dass ein Controller die Aktivität *create_case*, *order_translation*, *accept_offer*, *accept_delivery* oder *request_calculation* ausführt. Fehlerhafterweise wurden bei Testaufträgen, welche durch die Entwickler-Rolle durchlaufen wurden, alle Aktivitäten unter der Stelle Controller protokolliert, wodurch es sich nachweislich tatsächlich um Rauschdaten handelt.

7.4.3 Version 2.30

| <u>Action</u> | <u>Reason</u> |
|--|-----------------|
| Rename all Tasks cancel case executed by Controller to cancel case0. | Duplicated Task |
| Rename all Tasks cancel case executed by Customer to cancel case1. | Duplicated Task |
| Rename all Tasks do translation executed by Translator to do translation0. | Duplicated Task |
| Rename all Tasks do translation executed by Controller to do translation1. | Duplicated Task |

Tabelle 12: Aktionen des OPP - Version 2.30 - Parameter: 0

| <u>Action</u> | <u>Reason</u> |
|--|-----------------|
| Rename all Tasks cancel case executed by Controller to cancel case0. | Duplicated Task |
| Rename all Tasks cancel case executed by Customer to cancel case1. | Duplicated Task |
| Rename all Tasks do translation executed by Translator to do translation0. | Duplicated Task |
| Rename all Tasks do translation executed by Controller to do translation1. | Duplicated Task |

Tabelle 13: Aktionen des OPP - Version 2.30 - Parameter: 3

Das Ergebnis des Pre-Processing auf die Verlaufsdaten der Leginda Version 2.30 weist im Vergleich zur Version 2.20 einige Änderungen auf. Zunächst

fällt auf, dass kein Unterschied zwischen den Aktionen der Variante ohne Rauschdatenerkennung und den Aktionen der Variante mit Rauschdatenerkennung vorhanden ist. Ein Erklärungsversuch hierfür könnte darin liegen, dass das Fehlverhalten der Anwendung bei der Protokollierung erkannt und behoben wurde.

Seitens Leginda konnte diese Veränderung durch den Wegfall der Entwickler-Rolle in der Version 2.30 begründet werden, woraus implizit die Behebung des identifizierten Fehlers resultierte. Entgegen dem Erklärungsversuchs war der Fehler jedoch nicht im Vorfeld bekannt gewesen.

Eine Übereinstimmung der durchgeführten Aktionen über alle bisher durchgeführten Pre-Processing-Läufe kann in der Identifizierung doppelter Tasks bei *do_translation* und *cancel_case* gefunden werden.

7.4.4 Version 2.40

Die durchgeführten Aktionen beim Organizational Pre-Processing über die Verlaufsdaten der Version 2.40 entsprachen exakt denen der Version 2.30. Es konnten demzufolge in der Pre-Processing-Phase keine weiteren Workflow-Änderungen identifiziert werden.

Auffällig ist jedoch, dass auch hier die Tasks *do_translation* und *cancel_case* als doppelte Tasks interpretiert wurden.

7.4.5 Ergebnisse

Die Anwendung des hier entwickelten Organizational Pro-Processing auf Echtdaten konnte mehrere interessante Erkenntnisse liefern.

Es wurde gezeigt, dass der Ansatz entscheidende Beiträge für die Erkennung von Rauschdaten liefert. Da insbesondere Organisationswissen für diesen Vorgang verwendet wird, handelt es sich bei dem Ansatz um den derzeit einzigen, welcher in der Lage ist, diese Art von Rauschen zu erkennen. Bisher existiert kein weiteres Vorgehen, welches die Initiator-Information der Verlaufsdaten für diese Aufgabe verwendet. Weiterhin liefert der Pre-Processing Report durch die Logdaten-Analysen und die explizite Beschreibung der durchgeführten Aktionen wertvolle Informationen, welche die Fehlersuche unterstützen – demnach werden nicht nur die Logdaten bereinigt, sondern darüber hinaus Hinweise zur Fehlerdiagnose generiert.

Durch die getrennte Analyse einzelner Versionen konnten aus dem Pre-Processing Report ebenfalls Informationen über den Entwicklungsverlauf des Workflow-Systems abgeleitet werden. Der Übergang der Version 2.20 auf die Version 2.30 zeigte hier konkret den Wegfall von Rauschdaten.

In allen OPP (Organisational Pre-Processing)-Durchläufen wurden die Tasks *do_translation* und *cancel_case* als doppelt identifiziert, was sich nach Rücksprache mit der Leginda GmbH und einigen Nachforschungen als korrekt herausstellte und wie folgt erklärt werden kann.

Do_translation wurde einerseits von Übersetzern, andererseits von Controllern durchgeführt, woraus eine Umbenennung in *do_translation0* (Controller) und *do_translation1* (Übersetzer) resultierte. Es konnte geklärt werden, dass *do_translation1* den (korrekten) Fall kennzeichnet, in welchem der Übersetzer die Übersetzung vornimmt und ins System lädt. Im Gegensatz dazu kennzeichnet *do_translation0* ein Szenario, in welchem der Übersetzer die Übersetzung durchführt, diese jedoch (am System vorbei) an den Controller übermittelt, welcher die Übersetzung anschließend in das System lädt.

Cancel_case wurde in den Verlaufsdaten von Controllern und Kunden ausgeführt und demzufolge in *cancel_case0* (Kunde) und *cancel_case1* (Controller) umbenannt. Während *cancel_case0* einer Ablehnung des Übersetzungsangebots durch den Kunden entspricht (woraus natürlich auch ein Abbruch des Geschäftsvorfalles resultiert) handelt es sich bei *cancel_case1* um einen Abbruch durch einen Controller, was unterschiedliche Gründe haben kann. Beispielsweise könnte der Fall eintreten, dass kein Übersetzer für die Bearbeitung des Auftrags verfügbar ist.

Die Ausführungen machen deutlich, dass eine Unterscheidung zwischen doppelten und versteckten Tasks nicht trivial ist. Bei beiden dargestellten Fällen handelt es sich klarerweise um doppelte Tasks, da mit einer Bezeichnung zwei unterschiedliche Vorgänge dargestellt werden. Gleichzeitig könnte man jedoch auch von versteckten Tasks reden. Beispielsweise wird bei *do_translation* in jedem Fall eine Übersetzung vom Übersetzer erzeugt. Der Unterschied liegt lediglich in der Art und Weise, wie und durch wen diese Übersetzung in das System gelangt. Dies könnte also so interpretiert werden, dass *do_translation* den eigentlichen Übersetzungsvorgang bezeichnet, während es sich bei der Übertragung der Übersetzung ins System um einen versteckten Task handelt.

Unabhängig davon, ob der Sachverhalt als doppelter oder versteckter Task (oder beides) betrachtet wird, bleibt festzuhalten, dass kein weiterer bekannter Algorithmus in der Lage wäre eine Abgrenzung (z.B. wann wird *cancel_case0* gewählt und wann *cancel_case1*) im konkreten Fall durchzuführen, da in diversen Zuständen, wie beispielsweise vor der Beauftragung durch den Kunden

(siehe Prozessbeschreibung in Kapitel 7.2.2), die Entscheidung ausschließlich auf Basis der Initiator-Information getroffen werden kann.

7.5 Processing

7.5.1 Vorgehensweise

Im Folgenden wird das Processing einerseits auf die extrahierten, andererseits auf die durch das OPP bereinigten Verlaufsdaten angewendet, wobei unterschiedliche Varianten verwendet werden.

Wie bereits erwähnt kommen hierzu der Alpha Algorithmus und der Genetic Miner (Standardparametrisierung) zum Einsatz. Beide Methoden werden auf alle originalen Verlaufsdaten der drei Versionen angewendet. Weiterhin werden die bereinigten Verlaufsdaten der Version 2.20 (mit und ohne Rauschdatenerkennung) und der Version 2.40 (mit Rauschdatenerkennung) herangezogen. Im Falle der Versionen 2.30 und 2.40 ist eine Variante ausreichend, da aus dem OPP identische Resultate hervorgingen. Für die Analyse wurde die Version 2.40 gewählt, da diese eine größere Anzahl an Prozessinstanzen beinhaltet.

7.5.2 Version 2.20

Der Alpha Algorithmus führt bei Anwendung auf die unterschiedlichen Verlaufsdaten zu sehr unterschiedlichen Ergebnissen. Den abgeleiteten Petri-Netzen (Abb. 25, 26, 27) kann entnommen werden, dass die ursprünglichen Verlaufsdaten zu einem sehr übersichtlichen Modell führen, während die Anwendung des OOP die Komplexität des Modells (in der Version 2.20) enorm erhöht.

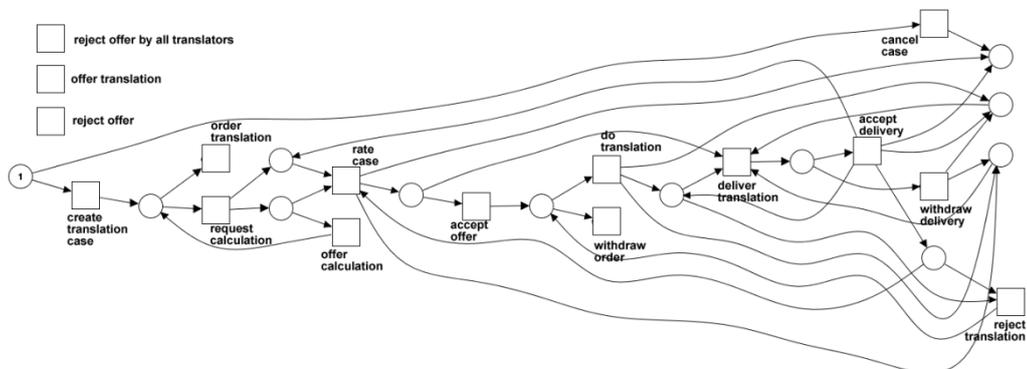


Abb. 25: Alpha Algorithmus - Leginda 2.20 – Original

Die größte Komplexitätssteigerung resultiert aus der Variante mit OOP und ohne Rauschdatenerkennung, da darin potentielle Rauschdaten im Sinne doppelter Tasks uminterpretiert werden, woraus eine wesentlich höhere Anzahl an Aktivitäten resultiert. Wird hingegen die Rauschdatenerkennung des OOP genutzt (hier mit einem Parameter von 3) beschränken sich die Uminterpretationen bzw. Umbenennungen von Aktivitäten auf die (hier) tatsächlich doppelten Tasks, was im Vergleich zur vorherigen Variante zu einer Komplexitätsreduktion führt.

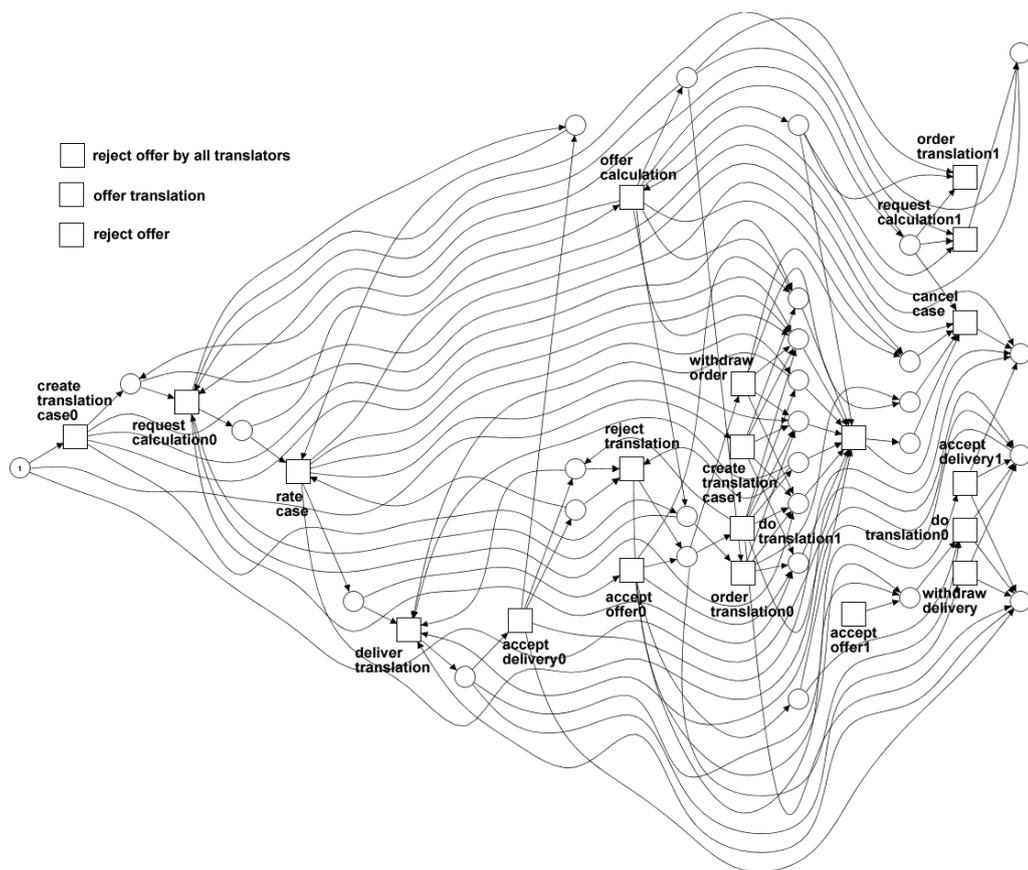


Abb. 26: Alpha Algorithmus - Leginda 2.20 - OPP, Noise: 0

Bei den abgeleiteten Petri-Netzen ist auffällig, dass alle die Aktivität *offer_translation* separat, also nicht im Modell verankert, darstellen, obwohl sie in einem Großteil der Prozessinstanzen vorhanden ist. Gleiches gilt für die Aktivitäten *reject_offer_by_all_translators* und *reject_offer*, allerdings treten diese verhältnismäßig selten auf.

Darüber hinaus fällt auf, dass mehrere Senken erkannt werden, die in der Realität nicht existieren (bspw. *do_translation* und *withdraw_delivery*). Auch andere Stellen innerhalb des Prozesses, wie beispielsweise *order_translation* ohne

Nachfolgeaktivität, entsprechen nicht den Erwartungen und insbesondere nicht den Verlaufsdaten.

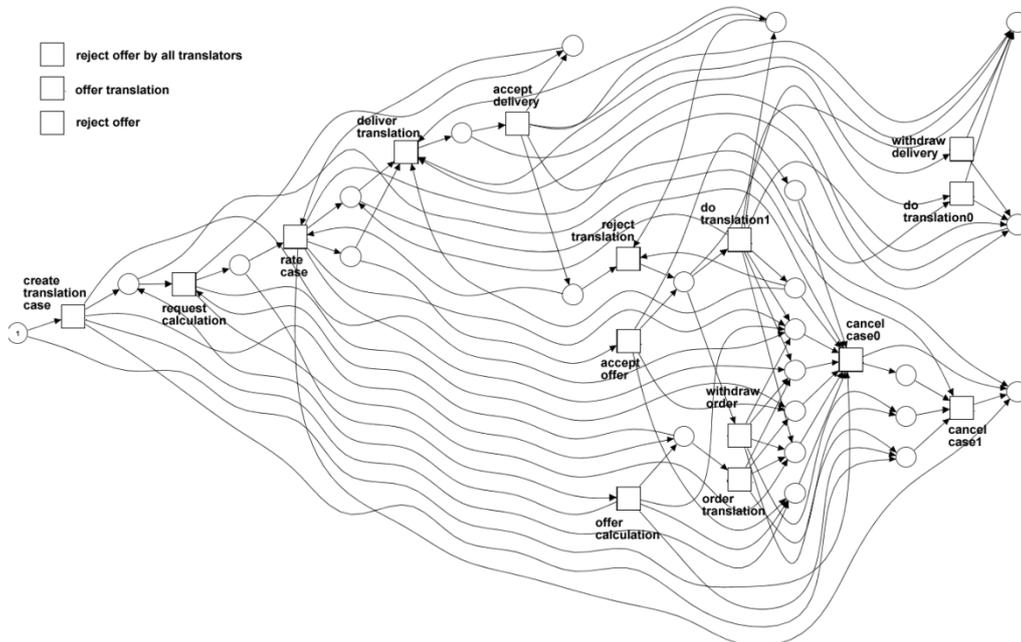


Abb. 27: Alpha Algorithmus - Leginda 2.20 - OPP, Noise: 3

Betrachtet man hingegen die durch den Genetic Miner abgeleiteten Modelle (Abb. 28, 29, 30), so gelangt man bei allen Verlaufsdaten zu erheblich weniger komplexen Resultaten. Auch hier werden Aktivitäten außerhalb des Prozesses modelliert, trotzdem ist der abgeleitete Prozess wesentlich genauer und spiegelt eher den realen Prozess wieder, als es die durch den Alpha Algorithmus abgeleiteten Modelle tun. Lediglich die Aktivität *withdraw_delivery* wird durchgängig separat modelliert.

Als Senke tritt stets wie erwartet nur *cancel_case (1)* auf.

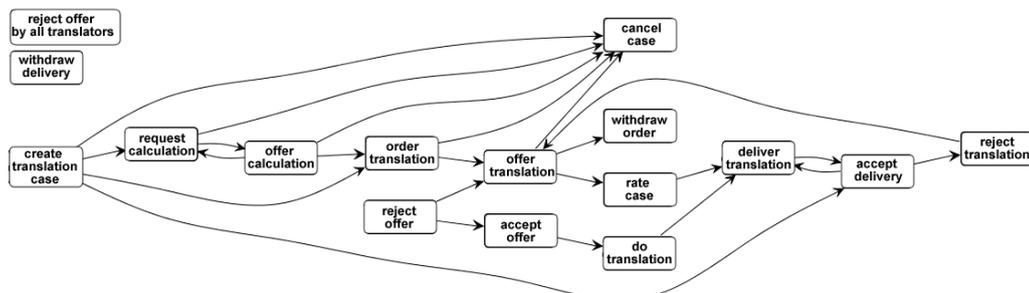


Abb. 28: Genetic Miner - Leginda 2.20 – Original

Besonders interessant ist das Ergebnis bei den Verlaufsdaten mit OPP ohne Rauschdatenkennung. In der oberen rechten Ecke scheint ein zweiter Prozess zu existieren, welcher die Testfälle durch die Developer-Roller repräsentiert. Im realen Einsatz könnte dieser Modellteil auf einen Fehler innerhalb des Prozesses bei der Ausführung oder der Protokollierung hinweisen und so eine konkrete Prüfung des Sachverhalts anstoßen.

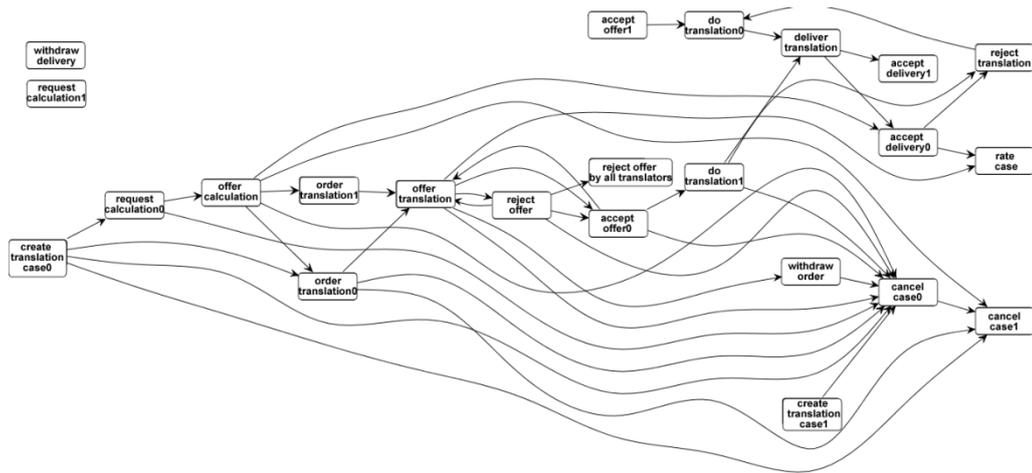


Abb. 29: Genetic Miner - Leginda 2.20 - OPP, Noise: 0

Das abgeleitete Modell aus den Verlaufsdaten mit OPP und mit Rauschdaten-erkennung kommt dem realen Prozess sehr nahe. Die Aktivitäten treten insgesamt in der korrekten Reihenfolge auf, jedoch sind einige Transitionen, wie beispielsweise von *accept_offer* zu *offer_translation*, unklar. Dies könnte evtl. auf einen Fehler hindeuten, was im Anschluss untersucht werden muss.

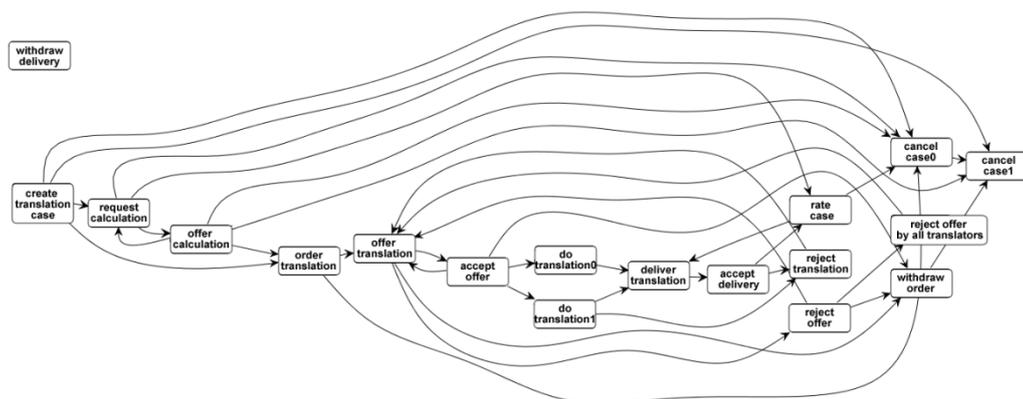


Abb. 30: Genetic Miner - Leginda 2.20 - OPP, Noise: 3

Da der Genetic Minder nach dem Prozessmodell mit dem besten Fitness-Wert sucht (diejenige Variante, die den Verlaufsdaten am besten gerecht wird), ist die separate Modellierung der Aktivität *withdraw_delivery* durch das einmalige Auftreten innerhalb der Leginda Version 2.20 zu erklären (siehe Tabelle 9).

Deutlich erkennbar sind insbesondere die alternativen Vorgehensweisen bei *do_translation* und *cancel_case*. Aus dem Modell geht die zusätzliche Information hervor, dass ausschließlich *cancel_case1* (Abbruch durch den Controller) zu einem tatsächlichen Abbruch des Geschäftsvorfalles führt (Senke), wohingegen *cancel_case0* (Ausführung durch Kunde) einer Angebotsablehnung entspricht (keine Senke).

7.5.3 Version 2.40

Die Anwendung des Alpha Algorithmus auf die Verlaufsdaten der Leginda Version 2.40 (Original und nach OPP) führt zu wesentlich komplexeren Modellen als es noch in der Version 2.20 der Fall war. Dies kann zwei mögliche Ursachen haben – einerseits könnte der Prozessablauf verändert worden sein, andererseits könnten seltene Prozesspfade, welche in der Version 2.20 nicht eingeschlagen wurden, zu diesem Resultat führen.

Die Modelle zeigen jedoch, dass die Anwendung des OPP nicht zwingend zu einer Steigerung der Modellkomplexität führen muss – dieses Ergebnis, welches aus dem vorherigen Kapitel hervorging, kann hier nicht bestätigt werden (siehe Abb. 31).

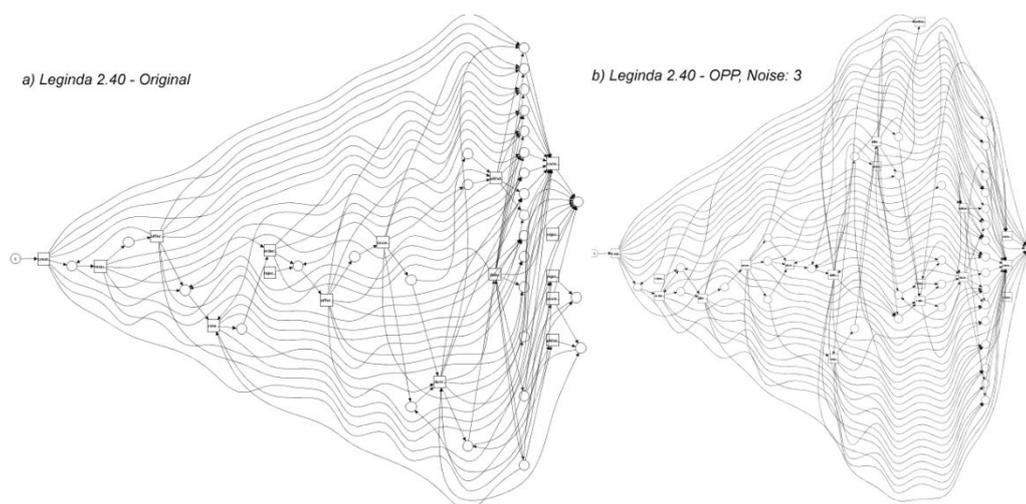


Abb. 31: Alpha Algorithmus - Leginda 2.40

Auffällig ist weiterhin, dass nun alle Aktivitäten korrekt in den Prozess modelliert werden konnten, was in der Version 2.40 nicht möglich war. Eine Begründung dafür könnte in der Behebung von Fehlern während der Protokollierung liegen.

Das Beispiel zeigt jedoch, dass der Alpha Algorithmus nur bedingt für eine Evaluation wie diese sinnvoll ist, da die Output-Modelle schnell sehr unübersichtlich werden, wodurch eine Analyse extrem aufwändig wird.

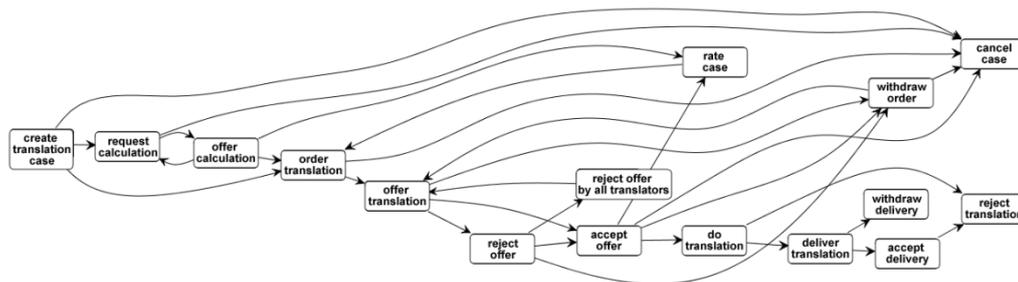


Abb. 32: Genetic Miner - Leginda 2.40 – Original

Erwartungsgemäß konnte der Genetic Miner jedoch erheblich bessere Ergebnisse liefern. Auch in der aktuellen Leginda Version sind die resultierenden Modelle sehr nahe an der Realität, spiegeln jedoch nicht immer die Verlaufsdaten wider.

Das Modell, welches aus den durch OPP bereinigten Verlaufsdaten abgeleitet wurde, zeigt erneut die Unterschiede zwischen *do_translation0* (Controller) und *do_translation1* (Übersetzer). Interessant ist vor allem, dass dem Modell zufolge Abhängigkeiten des nachfolgenden Verhaltens zu diesen Aktivitäten vorhanden sind. Demnach wird in der direkten Nachfolge, eine Übersetzung nur dann zurückgegeben (*reject_translation*), wenn sie durch einen Übersetzer ins System geladen wurde (*do_translation1*) – es existiert keine Verbindung zwischen *do_translation0* zu *reject_translation*. Diese Feststellung erscheint durch den aktuellen Kenntnisstand plausibel, da bei *do_translation0* der Controller selbst die Übersetzung ins System lädt und deshalb auch vor dem Abschluss von *do_translation0* darauf Einfluss nehmen kann. Es könnte in diesem Fall also sein, dass obwohl keine Verbindung von *do_translation0* zu *reject_translation* besteht, *reject_translation* als versteckter Task in *do_translation0* ausgeführt wird.

Auch im hier abgeleiteten Modell (Abb. 33) kommt der Unterschied zwischen *cancel_case0* (Kunde) und *cancel_case1* (Controller) hervor. Im Gegensatz zum vorherigen Kapitel scheint es nun jedoch so, als würde *cancel_case0* einer Angebotsablehnung mit anschließendem tatsächlichen Geschäftsfallabbruch

entsprechen, da *cancel_case0* hier als Senke modelliert ist. Es könnte sich demnach einerseits um einen Hidden Task, andererseits um eine Änderung des Prozessablaufs handeln.

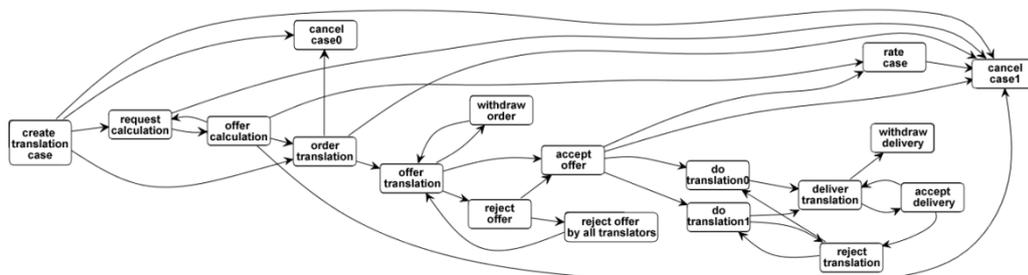


Abb. 33: Genetic Miner - Leginda 2.40 - OPP, Noise: 3

7.5.4 Ergebnisse

Da die Verlaufsdaten die Basis für das Processing bilden, hat das OPP signifikante Auswirkungen auf das abgeleitete Prozessmodell. Da doppelte Tasks a priori erkannt und behandelt wurden, können diese in geeigneter Weise im Processing berücksichtigt und im resultierenden Modell abgebildet werden. Da diese doppelten Tasks ausschließlich durch Organisationwissen identifiziert werden können, ist kein weiterer (bisher bekannter) Processing Algorithmus in der Lage, diese selbst zu erkennen. Durch die ebenfalls vorher entfernten Rauschdaten wurden weiterhin kritische Cases entfernt, wodurch Unstimmigkeiten im Hinblick auf die Zuordnung der doppelten Tasks bereinigt wurden.

Die höhere Anzahl an Aktivitäten, die durch das OPP generiert wird, führt, insbesondere beim Alpha Algorithmus, zu einer signifikanten Komplexitätssteigerungen, da folglich auch neue Transitionen abgeleitet werden. Es musste darüber hinaus festgestellt werden, dass der Alpha Algorithmus sehr schnell (also bereits bei wenigen Aktivitäten) sehr komplexe Modelle produziert, was mitunter darauf zurückzuführen ist, dass keine Mechanismen zur Rauschdaten-erkennung und ähnliches vorhanden sind. Demnach werden alle Verbindungen entsprechend des Algorithmus modelliert, selbst wenn sie nur einmal in den Verlaufsdaten vorkommen. Der Genetic Miner liefert hier erheblich intuitivere Ergebnisse. Über mehrere Generationen hinweg, wird dabei versucht, das am besten passende Modell (mit dem besten Fitness-Wert) zu den gegebenen Verlaufsdaten zu ermitteln. Der Vorteil dieses Vorgehens liegt darin, dass das Zielmodell verhältnismäßig übersichtlich bleibt und die breite Masse der Prozessinstanzen abbildet, jedoch kann die korrekte und vollständige Abbildung der Verlaufsdaten nicht garantiert werden.

Es muss darauf hingewiesen werden, dass das Ziel dieses Teilkapitels nicht in der Evaluation der Processing Algorithmen bzw. der Output Modelle lag, sondern vielmehr die Auswirkungen der bearbeiteten Verlaufsdaten auf das Processing untersucht werden sollten. Demnach kann festgehalten werden, dass das hier entwickelte und prototypisch implementierte Oranizational Pre-Processing zu genaueren Ergebnissen in der Processing-Phase führt, da eventuelle Fehler und doppelte Tasks in den Verlaufsdaten a priori behoben werden.

Für die Evaluation von Mining-Algorithmen und Anwendungen sei auf die Arbeiten von Rozinat et. al¹⁰⁰ und Weber¹⁰¹ verwiesen.

¹⁰⁰ Rozinat, A., Alves de Meideros, A.K., Günther, C.W., Weijters, A.J.M.M., van der Aalst, W.M.P.: Towards an Evaluation Framework for Process Mining Algorithms, In: BPM'07 Proceedings of the 2007 international conference on Business process management.

¹⁰¹ Weber, P.: Evaluation of Process Mining Tools and Algorithms, URL: http://www.cs.bham.ac.uk/~pxw869/papers/MSc/sem1_miniproj.pdf, Abrufdatum: 03.08.2011.

8 Zusammenfassung und Fazit

Nach einer Einführung in die Grundlagen des Process Mining, welche eine Literaturübersicht und eine daraus abgeleitete Historie beinhaltet, und einem State-of-the-Art des Process Mining in Verbindung mit Organisationswissen, wurde im Rahmen der Arbeit untersucht, inwiefern a priori vorhandenes Organisationswissen für das PM genutzt werden kann.

Es wurden Ansätze entwickelt, welche sich mit der Lösung der konkreten Problemstellungen innerhalb des Themengebietes befassen, unterschiedliche Sichtweisen behandeln und Aussagen über deren Umsetzbarkeit treffen.

Für diese Ansätze wurde ein Organisations-Metamodell entwickelt, welches einerseits den State-of-the-Art repräsentiert, andererseits die notwendigen Daten abbilden kann. Dieses Organisations-Metamodell dient als Grundlage für die Umsetzung der entwickelten Ansätze und wurde als ProM-Plugin prototypisch implementiert.

Zwei der entwickelten Ansätze – zur Erkennung doppelter Tasks und zur Erkennung von Rauschdaten – wurden formalisiert und ebenfalls prototypisch implementiert. Es entstand dadurch das Organizational Pre-Processing, welches als ProM-Plugin auf vorhandene Verlaufsdaten und ein Organisationsmodell angewendet werden kann und neue, bereinigte Verlaufsdaten produziert.

Eine Evaluation mit Echtdateien aus einem Workflow-System hat gezeigt, dass das Verfahren in der Lage ist, doppelte Tasks korrekt zu erkennen und Rauschdaten zu eliminieren. Darüber hinaus werden durch einen automatisch generierten Pre-Processing-Report wertvolle Informationen zu eventuellen Fehlern im Prozessablauf bzw. in der Protokollierung erzeugt, welche eine anschließende Überprüfung und Nachbesserung der entsprechenden Stellen anstoßen können.

Die resultierenden Verlaufsdaten wurden in ein Processing mit Alpha Algorithmus und Genetic Miner überführt, was aufgrund der a priori bekannten doppelten Tasks zu besseren (näher an der Realität gelegenen) Ergebnissen führte. Hieraus konnten weiterführende Informationen über den tatsächlichen Prozessablauf abgeleitet werden, was ohne das OPP nicht möglich gewesen wäre - dazu zählen beispielsweise Hinweise auf versteckte Tasks.

Die Evaluation hat ferner gezeigt, dass die Qualität der Ergebnisse entscheidend von der Korrektheit der Input-Daten – also Verlaufsdaten und Organisationsmodell – abhängig ist.

9 Ausblick

In der vorliegenden Arbeit wurden zwei der sechs entwickelten Ansätze zum verbesserten Process Mining unter Verwendung von Organisationswissen formalisiert, implementiert und evaluiert. Demnach ist es denkbar auch die übrigen Ansätze umzusetzen und auf ihre Tauglichkeit hin zu untersuchen. Im Gegensatz zur hier vorgestellten Umsetzung sind die übrigen Ansätze in die Phasen Processing und Post-Processing einzuordnen.

Die Arbeit konnte zeigen, dass a priori vorhandenes Organisationswissen in der Lage ist, den Mining-Prozess erheblich zu unterstützen. Da es sich hierbei um die erste bisher bekannte Arbeit in diesem speziellen Teilbereich des Process Mining handelt, wäre die Vertiefung der vorgestellten Ansätze, ebenso wie die Weiterentwicklung sinnvoll und denkbar.

In nachfolgenden Arbeiten sollte untersucht werden, inwiefern auch anderes, im Vorfeld vorhandenes Wissen, gewinnbringend im Process Mining eingesetzt werden kann. Vorstellbar wäre z.B. das Wissen über Daten, Wissen über den Prozess selbst und das Wissen über weitere Informationen, die mit dem Prozess in Verbindung stehen.

Eine Lücke in den aktuellen Forschungsanstrengungen besteht im Vergleich der einzelnen PM-Algorithmen. Es existiert bisher keine Arbeit, welche die vorhandenen Algorithmen aufgreift und im Detail evaluiert. Gerade daraus könnten sehr interessante Erkenntnisse darüber gewonnen werden, welche Ansätze am genauesten arbeiten, ob Ansätze existieren, die in bestimmten Branchen besonders effektiv sind, oder wie die Protokollierung von Informationssystemen verändert werden kann, um den Mining Prozess zu optimieren.

Abschließend wird festgehalten, dass sich die Forschungen und Entwicklungen im Bereich des Process Mining in vollem Gange befinden und noch sehr viel Potential für nachfolgende Forschungsanstrengungen und neue Ansätze vorhanden ist.

Anhang

Literaturüberblick – Process Mining

| ID | Autor | Erscheinungsjahr | Tiwari / Turner | Typ | Name des Papers |
|----|----------------------------|------------------|-----------------|---------|---|
| 1 | Cook und Wolf | 1995 | | Methode | Automating Process Discovery through Event-Data Analysis |
| 2 | Cook und Wolf | 1998 | x | Methode | Discovering models of software processes from event-based data |
| 3 | Cook und Wolf | 1998 | x | Methode | Event-based detection of concurrency |
| 4 | Herbst und Karagiannis | 1998 | x | Methode | Integrating machine learning and workflow management to support acquisition and adaptation of workflow models |
| 5 | Agrawal et al. | 1998 | x | Methode | Mining process models from workflow logs |
| 6 | Mannila und Rusakov | 2001 | x | Methode | Decomposition of event sequences into independent components |
| 7 | Adams et al. | 2001 | x | Methode | Mining for classes and patterns in behavioural data |
| 8 | Weijters und van der Aalst | 2001 | x | Methode | Process mining: discovering workflow models from event based data |
| 9 | Maruster et al. | 2002 | x | Methode | Automated discovery of workflow logs from hospital data |
| 10 | Hwang et al. | 2002 | x | Methode | On the discovery of process models from their instances |
| 11 | Maruster et al. | 2002 | x | Methode | Process mining: discovering direct successors in process logs |
| 12 | van der Aalst et al. | 2002 | x | Methode | Workflow mining: which processes can be rediscovered? |
| 13 | Chen und Yun | 2003 | x | Methode | Discovering process models from execution history by graph matching |
| 14 | Golani und Pinter | 2003 | x | Methode | Generating a process model from a process audit log |
| 15 | Schimm | 2003 | x | Methode | Mining most specific workflow models from event-based data |
| 16 | Weijters und van der Aalst | 2003 | x | Methode | Rediscovering workflow models from event-based data using little thumb |

| ID | Autor | Erscheinungsjahr | Tiwari / Turner | Typ | Name des Papers |
|-----------|--------------------------|-------------------------|------------------------|------------|--|
| 17 | de Medeiros et al. | 2003 | x | Methode | Workflow mining: current status and future directions |
| 18 | Zhang et al. | 2003 | x | Methode | Workflow process mining based on machine learning |
| 19 | Cook et al. | 2004 | x | Methode | Discovering models of behaviour for concurrent workflows |
| 20 | Hwang et al. | 2004 | x | Methode | Discovery of temporal patterns from process instances |
| 21 | Dongen und van der Aalst | 2004 | x | Methode | EMIT: a process mining tool |
| 22 | Hammori et al. | 2004 | x | Methode | Interactive workflow mining |
| 23 | Schimm | 2004 | x | Methode | Mining exact models of concurrent workflows |
| 24 | Greco et al. | 2004 | x | Methode | Mining expressive process models by clustering workflow traces |
| 25 | Dustdar et al. | 2004 | x | Methode | Mining of ad-hoc business processes with Team-Log |
| 26 | van der Aalst and Song | 2004 | x | Methode | Mining social networks: uncovering interaction patterns in business processes |
| 27 | Dongen und van der Aalst | 2004 | x | Methode | Multi-phase process mining: building instance graphs |
| 28 | de Medeiros et al. | 2004 | x | Methode | Process mining for ubiquitous mobile systems: an overview and a concrete algorithm |
| 29 | de Medeiros et al. | 2004 | | Methode | Process Mining: Extending the Alpha-algorithm to Mine Short Loops |
| 30 | de Medeiros et al. | 2004 | x | Methode | Using genetic algorithms to mine process models: representation, operators and results |
| 31 | van der Aalst | 2004 | x | Methode | Workflow mining: discovering process models from event logs |
| 32 | Dongen und van der Aalst | 2005 | x | Methode | A meta model for process mining data |
| 33 | van der Aalst et al. | 2005 | | Methode | Discovering Social Networks from Event Logs |
| 34 | van der Aalst et al. | 2005 | x | Methode | Genetic process mining |

| ID | Autor | Erscheinungsjahr | Tiwari / Turner | Typ | Name des Papers |
|-----------|--------------------------------|-------------------------|------------------------|------------|--|
| 35 | de Medeiros et al. | 2005 | x | Methode | Genetic process mining: a basic approach and its challenges |
| 36 | Greco et al. | 2005 | x | Methode | Mining and reasoning on workflows |
| 37 | Ly et al. | 2005 | | Methode | Mining Staff Assignment Rules from Event-Based Data |
| 38 | Gaaloul und Godart | 2005 | x | Methode | Mining workflow recovery from event based logs |
| 39 | Dongen und van der Aalst | 2005 | x | Methode | Multi-phase process mining: aggregating instance graphs into EPCs and Petri nets |
| 40 | Silva et al. | 2005 | | Methode | Probabilistic Workflow Mining |
| 41 | van der Aalst und de Medeiros | 2005 | x | Methode | Process mining and security: detecting process executions and checking process conformance |
| 42 | van der Aalst et al. | 2005 | x | Methode | Process mining and verification of properties: an approach based on temporal logic |
| 43 | Dongen et al. | 2005 | x | Methode | Verification of EPCs: using reduction rules and Petri nets |
| 44 | Sun et al. | 2006 | | Methode | Mining process models with non-free-choice constructs |
| 45 | Weijters et al. | 2006 | | Methode | Process Mining with the HeuristicsMiner Algorithm |
| 46 | van der Aalst und Günther | 2007 | | Methode | Finding Structure in Unstructured Processes: The Case for Process Mining |
| 47 | Günther und van der Aalst | 2007 | | Methode | Fuzzy Mining – Adaptive Process Simplification Based on Multi-Perspective Metrics |
| 48 | Wen et al. | 2007 | | Methode | Mining Invisible Tasks from Event Logs |
| 49 | de Medeiros et al. | 2007 | | Methode | Process Mining Based on Clustering: A Quest for Precision |
| 50 | van der Aalst et al. | 2007 | | Methode | Process Mining: A Two-Step Approach using Transition Systems and Regions |
| 51 | van der Werf et al. | 2008 | | Methode | Process Discovery using Integer Linear Programming |
| 52 | de Meideiros und van der Aalst | 2008 | | Methode | Process Mining towards Semantics |
| 53 | van der Aalst et al. | 2008 | | Methode | Process Mining: A Two-Step Approach to Balance Between Underfitting and Overfitting |

| ID | Autor | Erscheinungsjahr | Tiwari / Turner | Typ | Name des Papers |
|-----------|-----------------------------|-------------------------|------------------------|---------------|--|
| 54 | de Medeiros et al. | 2008 | | Methode | Semantic Process Mining Tools: Core Building Blocks |
| 55 | Song und van der Aalst | 2008 | | Methode | Towards Comprehensive Support for organizational Mining |
| 56 | Song et al. | 2008 | | Methode | Trace Clustering in Process Mining |
| 57 | Yaman et al. | 2009 | | Methode | A Context Driven Approach for Workflow Mining |
| 58 | Wen et al. | 2009 | | Methode | A Novel Approach for Process Mining Based on Event Types |
| 59 | Jagadeesh und van der Aalst | 2009 | | Methode | Context Aware Trace Clustering: Towards Improving Process Mining Results |
| 60 | Ferreira und Gillblad | 2009 | | Methode | Discovering Process models from unlabelled event logs |
| 61 | Rozinat et al. | 2009 | | Methode | Discovering Simulation Models |
| 62 | Ang et al. | 2009 | | Methode | Organizational Structure Mining Based on Workflow Logs |
| 63 | Goedertier et al. | 2009 | | Methode | Robust Process Discovery with Artificial Negative Events |
| 64 | Weijters und Ribeiro | 2010 | | Methode | Flexible Heuristic Miner (FHM) |
| 65 | Jagadeesh und van der Aalst | 2010 | | Methode | Trace Alignment in Process Mining: Opportunities for Process Diagnostics |
| 66 | Gombotz und Dustdar | 2004 | | Weiterführend | On Web Services Workflow Mining |
| 67 | van der Aalst | 2005 | | Weiterführend | Business Alignment – Using Process mining as a Tool for Delta Analysis and Conformance Testing |
| 68 | Dongen et al. | 2005 | x | Weiterführend | The ProM framework: a new era in process mining tool support |
| 69 | Rozinat und van der Aalst | 2006 | | Methode | Decision Mining in Business Processes |
| 70 | Rozinat et al. | 2007 | | Weiterführend | Towards an Evaluation Framework for Process Mining Algorithms |
| 71 | Goedertier et al. | 2008 | | Weiterführend | Process Mining as First-Order Classification Learning on Logs with Negative Events |

| ID | Autor | Erscheinungsjahr | Tiwari / Turner | Typ | Name des Papers |
|----|----------------------------|------------------|-----------------|---------------|---|
| 72 | Rozinat et al. | 2008 | | Weiterführend | Using Hidden Markov Models to Evaluate the Quality of Discovered Process Models |
| 73 | van der Aalst und Weijters | 2003 | | Überblick | Process Mining: A Research Agenda |
| 74 | van der Aalst et al. | 2003 | | Überblick | Workflow Mining: A survey of issues and approaches |
| 75 | Herbst and Karagiannis | 2004 | x | Überblick | Workflow mining with InWoLvE |
| 76 | Tiwari und Turner | 2008 | | Überblick | A review of business process mining: state-of-the-art and future trends |
| 77 | van der Aalst | 2008 | | Überblick | Decision Support Based on Process Mining |
| 78 | Dongen et al. | 2009 | | Überblick | Process Mining: Overview and Outlook of Petri Net Discovery Algorithms |

Tabelle 14: Literaturüberblick - Process Mining

Literaturanalyse – Process Mining

| ID | Sicht | | | Mining-Methode | | | | Modelltyp | | | | | | | |
|----|---------|--------------|------|----------------|-------------|--------|------------|-----------|------------|-----|-----|----------------------|------------|----------|-----------|
| | Prozess | Organisation | Fall | algorithmisch | statistisch | hybrid | Semantisch | EPK | Petri-Netz | FSM | CFG | Block-oriented Model | Soziogramm | Sonstige | Unbekannt |
| 1 | x | | | x | x | x | | | | x | x | | | | |
| 2 | x | | | | | x | | | | x | | | | | |
| 3 | x | | | x | | | | | | | | | | | x |
| 4 | x | | | | | x | | | | | | | | | x |

| ID | Sicht | | | Mining-Methode | | | | Modelltyp | | | | | | | |
|----|---------|--------------|------|----------------|-------------|--------|------------|-----------|------------|-----|-----|----------------------|------------|----------|-----------|
| | Prozess | Organisation | Fall | algorithmisch | statistisch | hybrid | Semantisch | EPK | Petri-Netz | FSM | CFG | Block-oriented Model | Soziogramm | Sonstige | Unbekannt |
| 5 | x | | | x | | | | | | | x | | | | |
| 6 | x | | | | | x | | | | | x | | | | |
| 7 | x | | | | | x | | | | | | | | | x |
| 8 | x | | | x | | | | | x | | | | | | |
| 9 | x | | | x | | | | | x | | | | | | |
| 10 | x | | | | | x | | | | | x | | | | |
| 11 | x | | | x | | | | | x | | | | | | |
| 12 | x | | | x | | | | | x | | | | | | |
| 13 | x | | | x | | | | | | | x | | | | |
| 14 | x | | | x | | | | | | | x | | | | |
| 15 | x | | | x | | | | | | | | x | | | |
| 16 | x | | | x | | | | | x | | | | | | |
| 17 | x | | | x | | | | | x | | | | | | |
| 18 | x | | | x | | | | | | | | | | x | |
| 19 | x | | | x | | | | | x | | | | | | |
| 20 | x | | | x | | | | | | | x | | | | |
| 21 | x | | | x | | | | | x | | | | | | |
| 22 | x | | | | | x | | | | | x | | | | |
| 23 | x | | | | | x | | | | | | | | | x |
| 24 | x | | | x | | | | | | | x | | | | |
| 25 | x | | | x | | | | | x | | | | | | |
| 26 | | x | | x | | | | | | | | | x | | |
| 27 | x | | | x | | | | x | x | | | | | | |
| 28 | x | | | x | | | | | x | | | | | | |
| 29 | x | | | x | | | | | x | | | | | | |
| 30 | x | | | x | | | | | x | | | | | x | |
| 31 | x | | | x | | | | | x | | | | | | |
| 32 | x | | | x | | | | x | x | | | | | | |
| 33 | | x | | | | x | | | | | | | x | | |
| 34 | x | | | x | | | | | x | | | | | | |
| 35 | x | | | x | | | | | x | | | | | | |
| 36 | x | | | x | | | | | | | x | | | | |
| 37 | | x | | | | x | | | | | | | | x | |
| 38 | x | | | x | | | | | | | x | | | | |

| ID | Sicht | | | Mining-Methode | | | | Modelltyp | | | | | | | |
|----|---------|--------------|------|----------------|-------------|--------|------------|-----------|------------|-----|-----|----------------------|------------|----------|-----------|
| | Prozess | Organisation | Fall | algorithmisch | statistisch | hybrid | Semantisch | EPK | Petri-Netz | FSM | CFG | Block-oriented Model | Soziogramm | Sonstige | Unbekannt |
| 39 | x | | | x | | | | x | x | | | | | | |
| 40 | x | | | | | x | | | | | x | | | | |
| 41 | x | | | x | | | | | x | | | | | | |
| 42 | x | | | x | | | | | x | | | | | | |
| 43 | x | | | x | | | | x | x | | | | | | |
| 44 | x | | | x | | | | | x | | | | | | |
| 45 | x | | | | | x | | | x | | | | | | |
| 46 | x | | | x | | | | | | | x | | | | |
| 47 | x | | | x | | | | | | | x | | | | |
| 48 | x | | | x | | | | | x | | | | | | |
| 49 | x | | | x | | | | | | | | | | | x |
| 50 | x | | | x | | | | | x | | | | | | |
| 51 | x | | | x | | | | | x | | | | | | |
| 52 | x | | | | | | x | | | | | | | | x |
| 53 | x | | | x | | | | | x | | | | | | |
| 54 | x | | | | | | x | | | | | | | | x |
| 55 | | x | | | | x | | | | | | | | | x |
| 56 | x | | | | | x | | | x | | | | | | |
| 57 | x | | | x | | | | | | | x | | | | |
| 58 | x | | | x | | | | | x | | | | | | |
| 59 | x | | | | | x | | | x | | | | | | |
| 60 | x | | | | | x | | | | x | | | | | |
| 61 | x | x | x | x | | | | | x | | | | | | |
| 62 | | x | | x | | | | | | | | | x | | |
| 63 | x | | | | | x | | | x | | | | | | |
| 64 | x | | | | | x | | | x | | | | | | |
| 65 | x | | | | | x | | | | | | | | x | |
| 66 | x | | | | | | | | | | | | | | |
| 67 | x | | | | | | | | | | | | | | |
| 68 | x | | | | | | | | | | | | | | |
| 69 | | | x | | | x | | | | | | | | x | |
| 70 | x | | | | | | | | | | | | | | |
| 71 | x | | | | | | | | | | | | | | |
| 72 | x | | | | | | | | | | | | | | |

| ID | Sicht | | | Mining-Methode | | | | Modelltyp | | | | | | | |
|----|---------|--------------|------|----------------|-------------|--------|------------|-----------|------------|-----|-----|----------------------|------------|----------|-----------|
| | Prozess | Organisation | Fall | algorithmisch | statistisch | hybrid | Semantisch | EPK | Petri-Netz | FSM | CFG | Block-oriented Model | Soziogramm | Sonstige | Unbekannt |
| 73 | x | | | | | | | | | | | | | | |
| 74 | x | | | | | | | | | | | | | | |
| 75 | x | | | | | | | | | | | | | | |
| 76 | x | | | | | | | | | | | | | | |
| 77 | x | | x | | | | | | | | | | | | |
| 78 | x | | | | | | | | x | | | | | | |

Tabelle 15: Literaturanalyse - Process Mining

Literaturverzeichnis

- Adams, N.M., Hand, D.J. and Till, R.J. (2001), Mining for classes and patterns in behavioural data, *Journal of the Operational Research Society*, Vol. 52, pp. 1017-24.
- Agrawal, R., Gunopulos, D. and Leymann, F. (1998), Mining process models from workflow logs, In: Schek, H.J. (Ed.), *Proceedings of the 6th International Conference on Extending Database Technology: Advances in Database Technology*, Springer Verlag, Heidelberg.
- Bissantz, N.; Hagedorn, J.: *Data Mining (Datenmustererkennung)* , In: *Wirtschaftsinformatik Volume 35* (1993).
- Braun, C.: *Modellierung der Unternehmensarchitektur - Weiterentwicklung einer bestehenden Methode und deren Abbildung in einem Meta-Modellierungswerkzeug*, Logos 2007.
- Braun, F., Kuhn, R.: *Der Einsatz von Process Mining Techniken in Revision und Compliance*, Deutsches Institut für Interne Revision e.V. / ProcessGold AG, <http://www.slideshare.net/ProcessGold/der-einsatz-von-process-mining-techniken-in-revision-und-compliance>, Abrufdatum: 12.05.2011.
- Chen, K.C.W. and Yun, D.Y.Y. (2003), Discovering process models from execution history by graph matching, in Liu, J. et al. (Eds), *IDEAL 2003*, Springer Verlag, Heidelberg, pp. 887-92.
- Cook, J.E. and Wolf, A.L. (1998a), Discovering models of software processes from event-based data, *ACM Transactions on Software Engineering and Methodology*, Vol. 7 No. 3, pp. 215-49.
- Cook, J.E. and Wolf, A.L. (1998b), Event-based detection of concurrency, *Proceedings of the 6th International Symposium on the Foundations of Software Engineering*, ACM Press, New York, NY.
- Cook, J.E., Du, Z., Liu, C. and Wolf, A.L. (2004), Discovering models of behaviour for concurrent workflows, *Computers in Industry*, Vol. 53, pp. 297-319.
- Cook, J.E., Wolf, A.L.: *Automating Process Discovery through Event-Data Analysis*, In: *ICSE '95 Proceedings of the 17th international conference on Software engineering*.

- Cook, J.E.; Wolf, A.L.: Discovering Models of Software Processes from Event-Based Data, In: ACM Transactions on Software Engineering and Methodology, Volume 7 Issue 3, Juli 1998.
- Cooper, H.M.: Synthesizing Research – A Guide for Literature Reviews. 3. Aufl., Thousand Oaks et al. 1998.
- de Medeiros, A., Karla, A., van der Aalst, W.M.P., Carlos, P.: Semantic Process Mining Tools: Core Building Blocks, In: International Conference on Semantic Computing 2008.
- de Medeiros, A.K.; van der Aalst, W.M.P.; Weijters; A.J.M.M.: Workflow mining: current status and future directions, in Meersman, R. et al. (Eds), CoopIS/DOA/ ODBASE 2003, Springer Verlag, Heidelberg.
- de Medeiros, A.K.; van Dongen, B.F.; van der Aalst, W.M.P.; Weijters, A.J.M.M.: Process mining for ubiquitous mobile systems: an overview and a concrete algorithm, in Baresi, L., Dustdar, S., Gall, H. and Materaseries, M. (Eds), Ubiquitous Mobile Information and Collaboration Systems (UMICS 2004), Springer Verlag, Heidelberg.
- de Medeiros, A.K.; Weijters, A.J.M.M.; van der Aalst, W.M.P.: Genetic process mining: a basic approach and its challenges, In: Bussler, C.; Haller, A. (Eds): Business Process Management Workshops: BPM 2005, Springer Verlag, Heidelberg.
- de Medeiros, A.K.; Weijters, A.J.M.M.; van der Aalst, W.M.P.: Using genetic algorithms to mine process models: representation, operators and results, Beta Working Paper Series, WP 124, Eindhoven University of Technology, Eindhoven.
- de Medeiros, A.K.A.; van der Aalst, W.M.P.: Process Mining towards Semantics, In: Advances in Web Semantics I.
- de Medeiros, A.K.A.; van Dongen, B.F.; van der Aalst, W.M.P.; Weijters, A.J.M.M.: Process Mining – Extending the a-Algorithm to Mine Short Loops,
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.5.7518&rep=rep1&type=pdf>, Abrufdatum: 12.05.2011.
- de Mereiros, A.K., Guzzo, A., Greco, G., van der Aalst, W.M.P, Weijters, A.J.M.M., van Dongen, B., Saccà, D.: Process Mining Based on Clustering: A Querst for Precision, In: BPM'07 Proceedings of the 2007 international conference on Business process management.

- Derungs, M.; Vogler, P.; Österle, H.: Metamodell Workflow. Bericht Nr. 3 (IM HSG/CC PSI/3), Kompetenzzentrum Prozess- und Systemintegration, Version 1.5, St. Gallen, 1996.
- Diaz-Bone, R.: Eine kurze Einführung in die sozialwissenschaftliche Netzwerkanalyse, In: Mitteilungen aus dem Schwerpunktbereich Methodenlehre, Heft Nr. 57, Berlin 2006, URL: http://www.rainer-diaz-bone.de/Diaz-Bone_Netzwerkanalyse.pdf, Abrufdatum: 09.06.2011.
- Dongen, B.F. and van der Aalst, W.M.P. (2004a), EMIT: a process mining tool, in Cortadelle, J. and Reisig, W. (Eds), 25th International Conference on Applications and Theory of Petri Nets (ICATPN 2004), Springer Verlag, Heidelberg.
- Dongen, B.F. and van der Aalst, W.M.P. (2004b), Multi-phase process mining: building instance graphs, in Atzeni, P., Chu, W., Lu, H., Zhou, S. and Ling, T.W. (Eds), Conceptual Modelling – ER 2004, Springer Verlag, Heidelberg.
- Dongen, B.F. and van der Aalst, W.M.P. (2005b), Multi-phase process mining: aggregating instance graphs into EPCs and Petri nets, paper presented at 2nd International Workshop on Applications of Petri Nets to Coordination, Workflow and Business Process Management, at the ICATPN, Miami, FL, June.
- Dongen, B.F., de Medeiros, A.K., Verbeek, H.M.W., Weijters, A.J.M.M. and van der Aalst, W.M.P. (2005b), The ProM framework: a new era in process mining tool support, in Ciardo, G. and Darondeau, P. (Eds), 26th International Conference on Applications and Theory of Petri Nets (ICATPN 2005), Springer Verlag, Heidelberg.
- Dongen, B.F., van der Aalst, W.M.P. and Verbeek, H.M.W. (2005a), Verification of EPCs: using reduction rules and Petri nets, in Pastor, O. and Cunha, J.F. (Eds), 17th International Conference on Advanced Information Systems Engineering (CAiSE 2005).
- Dongen, B.F.; van der Aalst, W.M.P.: A meta model for process mining data, Proceedings of the CAiSE'05 Workshops, Vol. 2.
- Dustdar, S., Hoffmann, T. and van der Aalst, W. (2004), Mining of ad-hoc business processes with TeamLog, Technical Report TUV-1841-2004-07, Technical University of Vienna, Vienna.
- Esswein, W.: Das Rollenmodell der Organisation: Die Berücksichtigung aufbauorganisatorischer Regelungen in Unternehmensmodellen, In: Augsbu-

- ger, W.; Sinz, E.J.: Bamberger Beiträge zur Wirtschaftsinformatik, Nr. 14, Bamberg 1992.
- Ferreira, D.R., Gillblad, D.: Discovering Process Models from Unlabelled Event Logs, In: BPM '09 Proceedings of the 7th International Conference on Business Process Management.
- Fettke, P.: State-of-the-Art des State-of-the-Art - Eine Untersuchung der Forschungsmethode „Review innerhalb der Wirtschaftsinformatik, Wirtschaftsinformatik 48 (2006) 4.
- Gaaloul, W. and Godart, C. (2005), Mining workflow recovery from event based logs, in van der Aalst, W.M.P. et al. (Eds), BPM 2005, Springer Verlag, Heidelberg.
- Galler, J.: Metamodelle des Workflow-Managements, In: IWi-Heft 121, Dezember 1995.
- Gao, A.; Yang, Y.; Zeng, M.; Zhang, J.-L.; Wang, Y.-W.: Organizational Structure Mining Based on Workflow Logs, In: International Conference on Business Intelligence and Financial Engineering 2009 (BIFE '09).
- Goedertier, S., Martens, D., Baesens, B., Haesen, R., Vanthienen, J.: Process Mining as First-Order Classification Learning on Logs with Negative Events, In: In A. ter Hofstede, B. Benatallah, and H.Y. Paik, editors, BPM 2007 International Workshops, volume 4928.
- Goedertier, S., Martens, D., Baesens, B., Haesen, R., Vanthienen, J.: Robust Process Discovery with Artificial Negative Events, In: The Journal of Machine Learning Research Volume 10, 12/1/2009.
- Golani, M. and Pinter, S. (2003), Generating a process model from a process audit log, in van der Aalst, W.M.P. (Ed.), Business Process Management, Springer Verlag, Heidelberg.
- Gombotz, R., Dustdar, S.: On Web Services Workflow Mining, In: Computers in Industry Volume 53, Issue 3
- Greco, G., Guzzo, A., Manco, G. and Sacca, D. (2005), Mining and reasoning on workflows, IEEE Transactions on Knowledge and Data Engineering, Vol. 17 No. 4, S. 519ff.

- Greco, G., Guzzo, A., Pontieri, L. and Sacca, D. (2004), Mining expressive process models by clustering workflow traces, in Dai, H., Srikant, R. and Zhang, C. (Eds), PAKDD 2004, Springer Verlag, Heidelberg, S. 52-62.
- Günther, C.W., van der Aalst, W.M.P.: Fuzzy Mining – Adaptive Process Simplification Based on Multi-Perspective Metrics, In: BPM'07 Proceedings of the 5th international conference on Business process management
- Günther, C.W. (Fluxicon Process Laboratories): XES – Standard Definition (Draft) Version 1, November 2009, S. 2.
- Hammori, M., Herbst, J. and Kleiner, N. (2004), Interactive workflow mining, in Desel, B.P. and Weske, M. (Eds), Business Process Management, Springer Verlag, Heidelberg, pp. 211-66.
- Hars, A., Zimmermann, V., Scheer, A.-W.: Entwicklungsrichtlinien für die computergestützte Modellierung von Aufbau- und Ablauforganisation, IWi-Heft 105, Dezember 1993.
- Heilmann, H.; Munz, D.: Die Integration der Aufbauorganisation in Workflow-Management-Systeme: Konzeption eines Metamodells und Entwicklung eines Prototyps, In: Software-Labor der Universität Stuttgart Bericht SL-3/97.
- Herbst, J. and Karagiannis, D. (1998), Integrating machine learning and workflow management to support acquisition and adaptation of workflow models, Proceedings of the 9th International Workshop on Database and Expert Systems Applications, IEEE Computer Society, Washington, DC.
- Herbst, J. and Karagiannis, D. (2004), Workflow mining with InWoLvE, Computers in Industry, Vol. 53, pp. 245-64.
- Herbst, J: Ein induktiver Ansatz zur Akquisition und Adaption von Workflow-Modellen, Tenea Verlag für Media, Berlin 2004.
- Hwang, S.Y. (2002), On the discovery of process models from their instances, Decision Support Systems, Vol. 34, pp. 41-57.
- Hwang, S.Y., Wei, C.P. and Yang, W.S. (2004), Discovery of temporal patterns from process instances, Computers in Industry, Vol. 53, pp. 345-64.
- Jagadeesh Chandra Bose, R.P., van der Aalst, W.M.P.: Context Aware Trace Clustering: Towards Improving Process Mining Results, In: Proceedings of the SIAM International Conference on Data Mining, SDM 2009, April 30 - May 2, 2009, Sparks, Nevada, USA 2009.

- Jagadeesh Chandra Bose, R.P., van der Aalst, W.M.P.: Trace Alignment in Process Mining: Opportunities for Process Diagnostics, In: BPM'10 Proceedings of the 8th international conference on Business process management.
- Jans, M., Lybaert, N., Vanhoof, K: Business Process Mining for Internal Fraud Risk Reduction: Result of a Case Study, URL: <http://www.ecmlpkdd2008.org/sites/ecmlpkdd2008.org/files/pdf/workshops/ipm/3.pdf>, Abrufdatum: 17.07.2011.
- Joosten, S.M.M.: WorkPAD: a Conceptual Framework for Process Analysis and Design, In: ACM Transactions on Office Information Systems 1996.
- Ly, L.T., Rinderle, S., Dadam, P., Reichert, M.: Mining Staff Assignment Rules from Event-Based Data, In: 3rd International Conference on Business Process Management 2005.
- Ly, L.T.: Process Mining – Bestehende Ansätze und weiterführende Aspekte, <http://dbis.eprints.uni-ulm.de/543/>, Zugang erforderlich, Abrufdatum: 14.10.2010.
- Mannila, H. and Rusakov, D. (2001), Decomposition of event sequences into independent components, Proceedings of the First SIAM International Conference on Data Mining, Chicago, 5-7 April.
- Maruster, L., van der Aalst, W.M.P., Weijters, T., van den Bosch, A. and Daelemans, W. (2002b), Automated discovery of workflow logs from hospital data, in Dousson, C., Höppner, F. and Quiniou, R. (Eds), Proceedings of the ECAI Workshop on Knowledge Discovery and Spatial Data, pp. 78-84.
- Maruster, L., Weijters, A.J.M.M., van der Aalst, W.M.P. and van den Bosch, A. (2002a), Process mining: discovering direct successors in process logs, Proceedings of the 5th International Conference on Discovery Science (Discovery Science 2002), Springer Verlag, Berlin.
- Minseok Song and Wil M.P. van der Aalst: Towards Comprehensive Support for organizational Mining, In: Decision Support Systems Volume 46 Issue 1, December, 2008.
- Petersohn, H.: Data Mining: Verfahren, Prozesse, Anwendungsarchitektur; Oldenbourg Wissenschaftsverlag 2005.

- Rosemann, M.; zur Mühlen, M.: Modellierung der Aufbauorganisation in Workflow-Management-Systemen: Kritische Bestandsaufnahme und Gestaltungsvorschläge, <http://www.b2b-workflow.de/Publications/PDF/MIRO.MIZU-EMISA%281997%29.pdf>, Abrufdatum: 14.06.2010.
- Rozinat, A., de Medeiros, A.K., Günther, C.W., Weijters, A.J.M.M., van der Aalst, W.M.P.: Towards an Evaluation Framework for Process Mining Algorithms, In: BPM'07 Proceedings of the 2007 international conference on Business process management.
- Rozinat, A., Mans, R.S., Song, M., van der Aalst, W.M.P.: Discovering Simulation Models, In: Journal of Information Systems, Volume 34 Issue 3, May 2009.
- Rozinat, A., van der Aalst, W.M.P.: Decision Mining in Business Processes, Eindhoven University of Technology, März 2006, URL: <http://wwwis.win.tue.nl/~wvdaalst/publications/p312.pdf>, Abrufdatum: 10.05.2011.
- Rozinat, A., Veloso, M., van der Aalst, W.M.P.: Using Hidden Markov Models to Evaluate the Quality of Discovered Process Models, In: Proceedings of Induction of Process Models IPM workshop at ECML PKDD 2008.
- Russell, N., ter Hofstede, A.H.M., van der Aalst, W.M.P, Mulyar, N.: Workflow Control-Flow Patterns – A Revised View, <http://www.workflowpatterns.com/documentation/documents/BPM-06-22.pdf>, Abrufdatum: 12.05.2011.
- Scheer, A.-W.: Architektur integrierter Informationssysteme. Springer, Berlin 1992.
- Scheer, A.-W.: ARIS - Modellierungsmethoden, Metamodelle, Anwendungen, Vierte Auflage, Springer 2001.
- Schimm, G. (2003), Mining most specific workflow models from event-based data, In: van der Aalst, W.M.P. (Ed.), Business Process Management, Springer Verlag, Heidelberg.
- Schimm, G. (2004), Mining exact models of concurrent workflows, Computers in Industry, Vol. 53, pp. 265-81.

- Silva, R., Zhang, J., Shanahan, J.G.: Probabilistic Workflow Mining, In: KDD '05 Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining.
- Social Network Analysis – Wikipedia, URL: http://en.wikipedia.org/wiki/Social_network_analysis_software, Abrufdatum: 09.06.2011.
- Song, M., Günther, C.W., van der Aalst, W.M.P.: Trace Clustering in Process Mining, In: In Proceedings of Business Process Management Workshops 2008.
- Song, M., van der Aalst, W.M.P.: Towards Comprehensive Support for Organizational Mining, Eindhoven University of Technology, Juni 2008, URL: <http://is.tm.tue.nl/staff/wvdaalst/publications/p484.pdf>, Abrufdatum: 10.04.2011.
- Sun, J; Wand, J; Wen, L, van der Aalst, W.M.P.: Mining process models with non-free-choice constructs, In: Data Mining and Knowledge Discovery, Volume 15 (2007).
- Tiwari, A., Turner, C.J.: A review of Business Process Mining – State-of-the-Art and Future Trends, In: Business Process Management Journal, Vol. 14.
- van der Aalst, W.M.P. (2004b), Workflow mining: discovering process models from event logs, IEEE Transactions on Knowledge and Data Engineering, Vol. 16 No. 9, pp. 1128-42.
- van der Aalst, W.M.P., de Beer, H.T. and van Dongen, B.F. (2005b), Process mining and verification of properties: an approach based on temporal logic, in Meersman, R. and Tari, Z. (Eds), On the Move to Meaningful Internet Systems 2005, Springer Verlag, Heidelberg.
- van der Aalst, W.M.P., de Medeiros, A.K. (2005), Process mining and security: detecting process executions and checking process conformance, Electronic Notes in Theoretical Computer Science, Vol. 121, pp. 3-21.
- van der Aalst, W.M.P., de Medeiros, A.K. and Weijters, A.J.M.M. (2005a), Genetic process mining, in Ciardo, G. (Ed.), Applications and Theory of Petri Nets, Springer Verlag, Heidelberg.
- van der Aalst, W.M.P., Günther, C.W., Finding Structure in Unstructured Processes: The Case for Process Mining, In: Seventh International Conference on Application of Concurrency to System Design 2007.

- van der Aalst, W.M.P., Rubin, V., van Dongen, B.F., Kindler, E., Günther, C.W.: Process Mining: A Two-Step Approach to Balance Between Underfitting and Overfitting, In: Process Mining: A Two-Step Approach to Balance Between Underfitting and Overfitting Volume 9, Number 1.
- van der Aalst, W.M.P., Rubin, V., van Dongen, B.F., Kindler, E., Günther, C.W.: Process Mining: A Two-Step Approach using Transition Systems and Regions, In: Software and Systems Modeling Volume 9, Number 1.
- van der Aalst, W.M.P., Song, M., Mining social networks: uncovering interaction patterns in business processes, in Desel, J., Pernici, B. and Weske, M. (Eds), International Conference on Business Process Management (BPM 2004), Springer Verlag, Berlin.
- van der Aalst, W.M.P., van Dongen, B.F., Herbst, J., Maruster, L., Schimm, G., Weijters, A.J.M.M.: Workflow Mining: A survey of issues and approaches, In: Data & Knowledge Engineering Volume 47 Issue 2, November 2003.
- van der Aalst, W.M.P., Weijters, A.J.M.M. and Maruster, L. (2002), Workflow mining: which processes can be rediscovered?, Beta Working Paper Series, WP 75, Eindhoven University of Technology, Eindhoven.
- van der Aalst, W.M.P., Weijters, A.J.M.M.: Process Mining – A Research Agenda, Eindhoven University of Technology, Juli 2003, URL: <http://tmitwww.tm.tue.nl/staff/wvdaalst/Publications/p219.pdf>, Abrufdatum: 10.05.2011.
- van der Aalst, W.M.P.: Business Alignment – Using Process mining as a Tool for Delta Analysis and Conformance Testing, In: Requirements Engineering, Volume 10, Springer 2005.
- van der Aalst, W.M.P.: Decision Support Based on Process Mining, In: Burstein, F; Holsapple C.W.: Handbook on Decision Support Systems 1 – Basic Themes, International Handbooks on Information Systems 2008.
- van der Aalst, W.M.P.: Process Mining, In: Liu, Ling; Özsu, M.T.: Encyclopedia of Database Systems, Springer 2009.
- van der Aalst, W.M.P.; Reijers, H.A.; Song, M.: Discovering Social Networks from Event Logs, In: Computer Supported Cooperative Work (CSCW) Volume 14, Number 6.

- van der Aalst, W.M.P: Business Alignment: Using Process Mining as a Tool for Delta Analysis, In: CAiSE Workshops (2), 2004.
- van der Aalst: Decision Support Based on Process Mining, In: International Handbooks on Information Systems, 2008, IV.
- van Dongen, B.F., de Medeiros, A.K., Wen, L.: Process Mining: Overview and Outlook of Petri Net Discovery Algorithms, In: Lecture Notes in Computer Science, 2009, Volume 5460/2009.
- Weber, P.: Evaluation of Process Mining Tools and Algorithms, URL: http://www.cs.bham.ac.uk/~pxw869/papers/MSc/sem1_miniproj.pdf, Abrufdatum: 03.08.2011.
- Wedekind, E.E.: Informationsmanagement in der Organisationsplanung, Deutscher Universitätsverlag.
- Weijters, A.J.M.M. and van der Aalst, W.M.P. (2001), Process mining: discovering workflow models from event based data, in Kröse, B., Rijke, M., Schreiber, G. and Someren, M. (Eds), Proceedings of the 13th Belgium-Netherlands Conference on Artificial Intelligence, Amsterdam, October.
- Weijters, A.J.M.M. and van der Aalst, W.M.P. (2003), Rediscovering workflow models from event-based data using little thumb, Integral Computer-aided Engineering, Vol. 10 No. 2, pp. 151-62.
- Weijters, A.J.M.M., Ribero, J.T.S., Flexible Heuristics Minter (FHM), In: BETA publicatie : working papers, No. 334.
- Weijters, A.J.M.M., van der Aalst, W.M.P., de Medeiros, A.K.: Process Mining with the HeuristicsMiner Algorithm.
- Wen, L., Wang, J., Sun, W.: Mining Invisible Tasks from Event Logs, In: Lecture Notes in Computer Science, 2007, Volume 4505/2007
- Wen, L., Wang, J., van der Aalst, W.M.P, Wang, Z. Sun, J.: A Novel Approach for Process Mining Based on Event Types, In: Journal of Intelligent Information Systems Volume 32 Issue 2.
- Werf, J.M., Dongen, B.F., Hurkens, C.A., Serebrenik, A.: Process Discovery Using Integer Linear Programming, In: PETRI NETS '08 Proceedings of the 29th international conference on Applications and Theory of Petri Nets.

- Wil M.P. van der Aalst, Hajo A. Reijers, Minseok Song: Discovering Social Networks from Event Logs, In: Computer Supported Cooperative Work (CSCW) Volume 14, Number 6, 549-593, DOI: 10.1007.
- Wilder, T., Hess, T.: Forschungsmethoden der Wirtschaftsinformatik – Eine empirische Untersuchung, In: Wirtschaftsinformatik 49.
- Wolf, A.L., Rosenblum, D.S.: A Study in Software Process Data Capture and Analysis, In: Proceedings of the Second International Conference on the Software Process (1993), URL: <http://users.ece.utexas.edu/~perry/education/SE-Intro/wolf-rosenblum.pdf>, Abrufdatum: 15.08.2011.
- Yaman, F., Oates, T., Burstein, M.: A Context Driven Approach for Workflow Mining, In: IJCAI'09 Proceedings of the 21st international joint conference on Artificial intelligence.
- Zhang, S.H., Gu, N., Lian, J.X. and Li, S.H. (2003), Workflow process mining based on machine learning, Proceedings of the Second International Conference on Machine Learning and Cybernetics, IEEE Computer Society, Washington, DC.