プロセスマイニング マニフェスト

(最終版)

Wil van der Aalst, Arya Adriansyah, Ana Karla Alves de Medeiros, Franco Arcieri, Thomas Baier, Tobias Blickle, Jagadeesh Chandra Bose, Peter van den Brand, Ronald Brandtjen, Joos Buijs, Andrea Burattin, Josep Carmona, Malu Castellanos, Jan Claes, Jonathan Cook, Nicola Costantini, Francisco Curbera, Ernesto Damiani, Massimiliano de Leoni, Pavlos Delias, Boudewijn van Dongen, Marlon Dumas, Schahram Dustdar, Dirk Fahland, Diogo R. Ferreira, Walid Gaaloul, Frank van Geffen, Sukriti Goel, Christian Günther, Antonella Guzzo, Paul Harmon, Arthur ter Hofstede, John Hoogland, Jon Espen Ingvaldsen, Koki Kato, Rudolf Kuhn, Akhil Kumar, Marcello La Rosa, Fabrizio Maggi, Donato Malerba, Ronny Mans, Alberto Manuel, Martin McCreesh, Paola Mello, Jan Mendling, Marco Montali, Hamid Motahari Nezhad, Michael zur Muehlen, Jorge Munoz-Gama, Luigi Pontieri, Joel Ribeiro, Anne Rozinat, Hugo Seguel Pérez, Ricardo Seguel Pérez, Marcos Sepúlveda, Jim Sinur, Pnina Soffer, Minseok Song, Alessandro Sperduti, Giovanni Stilo, Casper Stoel, Keith Swenson, Maurizio Talamo, Wei Tan, Chris Turner, Jan Vanthienen, George Varvaressos, Eric Verbeek, Marc Verdonk, Roberto Vigo, Jianmin Wang, Barbara Weber, Matthias Weidlich, Ton Weijters, Lijie Wen, Michael Westergaard, and Moe Wynn

IEEE プロセスマイニングタスクフォース (IEEE Task Force on Process Mining) ⋆
http://www.win.tue.nl/ieeetfpm/

要約 プロセスマイニング (process mining) 技術を用いると、今日の情報システムで普通に入手できる**イベントログから知識を抽出する**ことができる。プロセスマイニング技術はさまざまなアプリケーション領域において**プロセスを発見、監視、改善する**ための新しい手段となる。プロセスマイニングが興味を持たれるのには二つの主な要因がある。ひとつは、より多くのイベントが記録されるようになってきたため、プロセスの履歴について詳細な情報が分かるようになったことである。もうひとつは、競争が激しく急速に変化する環境において、ビジネスプロセスを改善しサポートすることが求められていることである。本マニフェストは IEEE プロセスマイニングタスクフォースが作成したもので、プロセスマイニングに関する議論が活発になることを目指している。また本マニフェストが指針と重要な課題 (チャレンジ) を明らかにして、**ソフトウェア開発者**、**科学者**、**コンサルタント**、**経営者**、**エンドユーザ**のための**ガイド**となるよう期待している。目標は新しいツールであるプロセスマイニングの成熟度を上げ、(再) 設計、制御、および運用業務プロセスのサポートを改善することである。

1 IEEE プロセスマイニングタスクフォース

マニフェストとは、集団による “原則と意図の公開宣言” である。本マニフェストは *IEEE* **プロセスマイニングタスクフォース**のメンバーと支持者が執筆した。本タスクフォースの目標はプロセスマイニング (process mining) の研究、開発、教育、実装、進化、理解を促進することである。

プロセスマイニングはまだ若い研究分野で、計算知能やデータマイニングの分野と、プロセスモデリング/分析の分野にまたがる。プロセスマイニングのアイデアは、 現状の (情報) システムで既に利用可能な**イベントログから知識を抽出して**、(想定したプロセスではない) **実際のプロセスを発見、監視、改善する**、というものである (図 1 を参照)。プロセスマイニングには、(自動) プロセス発見 ((automated) process discovery; イベントログからプロセスモデルを抽出する)、適合性検査 (conformance checking; モデルとログを比較し違いを監視する)、ソーシャルネットワーク/組織マイニング、シミュレーションモデルの自動構築、モデル拡張、モデル修正、事例予測、履歴ベースの推奨 (recommendation)、が含まれる。

⋆ 初版は *BPM 2011 Workshops proceedings*, Lecture Notes in Business Information Processing, Springer-Verlag, 2011 に掲載された。

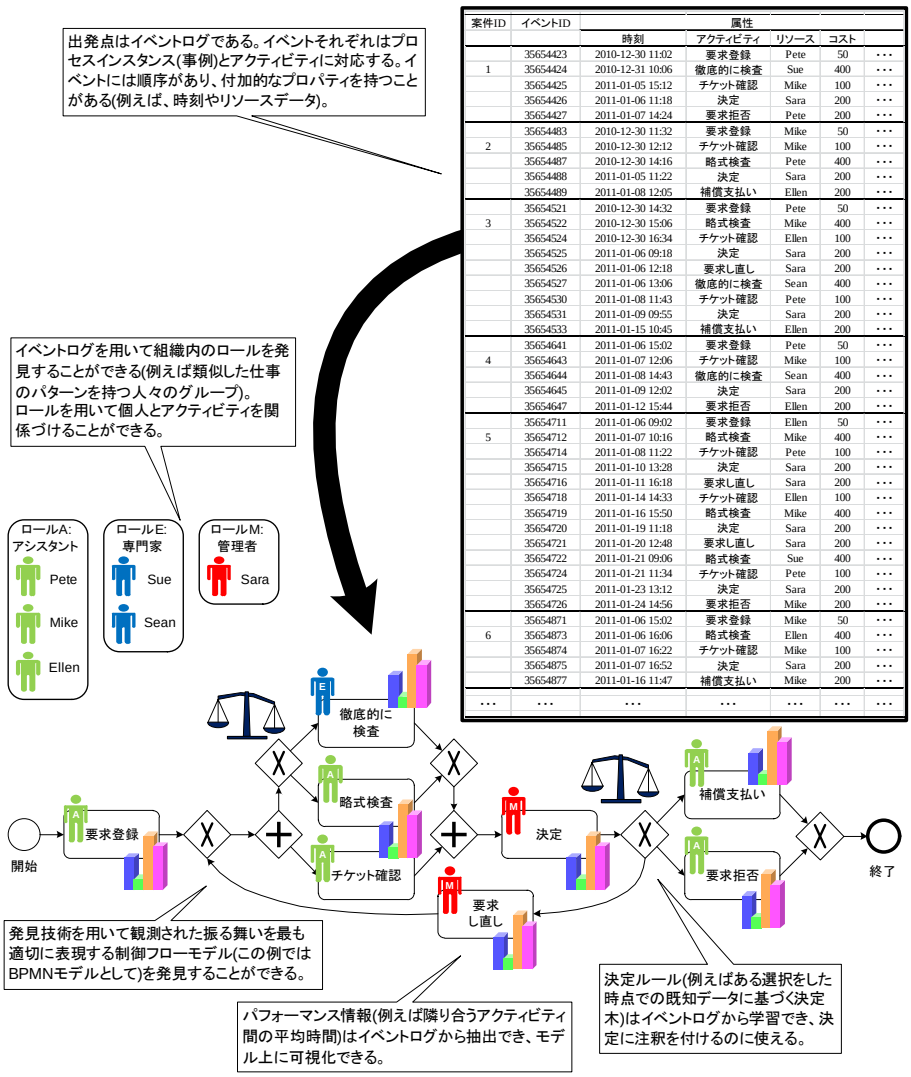

図 1. プロセスマイニング技術を用いてイベントログから知識を抽出し、プロセスを発見、監視、改善できる。

プロセスマイニングはデータマイニングと業務プロセスモデリング/分析との重要な架け橋になる。*Business Intelligence* (BI) の傘下に多くの業界用語が作られてきたが、中には割合と単純なレポート作成やダッシュボードツールも含まれている。*Business Activity Monitoring* (BAM) は業務プロセスのリアルタイム監視を実現する技術である。*Complex Event Processing* (CEP) は、リアルタイムに大量のイベントを処理することで、ビジネスを監視し導き最適化するのに用いる技術である。*Corporate Performance Management* (CPM) は、プロセスや組織のパフォーマンスを測定するための別の業界用語である。また管理向けの取り組みとしては、継続的なプロセス改善 (CPI)、業務プロセス改善 (BPI)、総合的品質管理 (TQM)、シックスシグマなどがある。これらの取り組みは更なる改善が可能かどうかを確認するためにプロセスを “顕微鏡下に置く” という点が共通している。プロセスマイニングは CPM、BPI、TQM、シックスシグマなどの実現技術である。

BI ツールとシックスシグマや TQM など管理手法が運用パフォーマンスの改善、例えばフロー時間と欠陥の削減を目指すのに対し、組織はまた、**コーポレートガバナンス、リスク管理、コンプライアンス**にも重点を置いている。サーベンスオクスリー法 (SOX) および新 BIS 規制などの法律は、コンプライアンスの問題に焦点を当てている。プロセスマイニング技術は組織のコアプロセスにおいて、より厳格にコンプライアンスを確認し情報の妥当性と信頼性を確実にするための手段となる。

この 10 年間ほどでイベントデータが容易に入手できるようになり、プロセスマイニング技術が成熟してきた。また上記のように、プロセス改善に関わる管理のトレンド (例えば、シックスシグマ、TQM、CPI、CPM) とコンプライアンス (SOX、BAM など) にプロセスマイニングを有効活用できる。幸いにもプロセスマイニングのアルゴリズムはさまざまな学術的システムや商用システムで実装されている。今日、プロセスマイニングに取り組んでいる研究者の積極的なグループがあり、ビジネスプロセスマネジメント (BPM) 研究の “ホットトピック” の一つとなっている。また、産業界もプロセスマイニングに大きな関心をもっており、ますます多くのソフトウェアベンダーが自社のツールにプロセスマイニング機能を追加している。プロセスマイニング機能を備えたソフトウェア製品の例として次のようなものがあげられる: ARIS Process Performance Manager (Software AG), Comprehend (Open Connect), Discovery Analyst (StereoLOGIC), Flow (Fourspark), Futura Reflect (Futura Process Intelligence), Interstage Automated Process Discovery (Fujitsu), OKT Process Mining suite (Exeura), Process Discovery Focus (Iontas/Verint), ProcessAnalyzer (QPR), ProM (TU/e), Rbminer/Dbminer (UPC), Reflect|one (Pallas Athena)。このようなログベースのプロセス分析への関心の高まりがプロセスマイニングタスクフォースの設立のきっかけとなった。

本タスクフォースは 2009 年に、Institute of Electrical and Electronic Engineers (IEEE) の Computational Intelligence Society (CIS) の下部組織である Data Mining Technical Committee (DMTC) に関連して設立された。現在のタスクフォースのメンバーは次の団体に及ぶ: **ソフトウェアベンダー** (例えば Pallas Athena, Software AG, Futura Process Intelligence, HP, IBM, Infosys, Fluxicon, Businesscape, Iontas/Verint, Fujitsu, Fujitsu Laboratories, Business Process Mining, Stereologic)、**コンサルタント会社/エンドユーザ** (例えば ProcessGold, Business Process Trends, Gartner, Deloitte, Process Sphere, Siav SpA, BPM Chili, BWI Systeme GmbH, Excellentia BPM, Rabobank)、**研究機関** (例えば TU/e, University of Padua, Universitat Politècnica de Catalunya, New Mexico State University, IST - Technical University of Lisbon, University of Calabria, Penn State University, University of Bari, Humboldt-Universität zu Berlin, Queensland University of Technology, Vienna University of Economics and Business, Stevens Institute of Technology, University of Haifa, University of Bologna, Ulsan National Institute of Science and Technology, Cranfield University, K.U. Leuven, Tsinghua University, University of Innsbruck, University of Tartu)。

タスクフォースの具体的な目的は次の通りである:

- エンドユーザ、開発者、コンサルタント、経営者、研究者から、プロセスマイニングの最先端技術に注目してもらうこと、
- プロセスマイニング技術、ツールの利用を促進し、新たなアプリケーションのきっかけを作ること、
- イベントデータのログに関する標準化に携わること、
- チュートリアル、セッション、ワークショップ、パネルを主催すること、
- 論文、本、ビデオ、ジャーナルの特別号を出すこと。

2009 年の設立以来、上記の目的に関連するさまざまな活動が行われている。例えば、いくつかのワークショップや特別トラックがタスクフォースによって (共同) 主催された (Business Process Intelligence ワークショップ (BPI'09、BPI'10、BPI'11) や、IEEE 会議での特別トラック (CIDM'11))。また、チュートリアル (例えば WCCI'10 と PMPM'09)、サマースクール (ESSCaSS'09、 ACPN'10、CICH'10、等)、ビデオ (www.processmining.org)、出版物 (Springer から出版されたプロセスマイニングの最初の本[1] を含む) を介して知識を広めてきた。また、本タスクフォースは第 1 回 Business Process Intelligence Challenge (BPIC'11) を (共同) 開催した。これは参加者が大規模・複雑なイベントログから有益な知識を抽出するコンテストであった。2010 年には、タスクフォースは標準ログフォーマットで拡張可能な *XES* (www.xes-standard.org) の標準化も行い、*OpenXES library* (www.openxes.org) や、ProM、XESame、Nitro などのツールでサポートされている。

http://www.win.tue.nl/ieeetfpm/ にはタスクフォースの活動の詳細が載っているので参照されたい。

[1] W.M.P. van der Aalst. Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer-Verlag, Berlin, 2011. http://www.processmining.org/book/

2 プロセスマイニング: 最先端の技術

情報システムやコンピュータに依存しているシステムの拡張傾向は、よくムーアの法則によって特徴付けられる。Intel の共同創設者である Gordon Moore は 1965 年に、集積回路内のコンポーネント数が毎年倍増することを予測した。過去 50 年間において、少し遅いペースではあるが確かに指数関数的に成長している。これらの進歩により、“デジタルユニバース” (すべてのデータが電子的に保存・やりとりされる) が目覚ましく成長した。また、デジタルと現実の世界がますます緊密に提携してきている。

デジタルユニバースは組織内のプロセスとうまく提携しているため、**イベント**を記録して分析するのが容易になる。イベントは例えば ATM からの現金引き出し、X 線機器の調整、運転免許証の申請、納税申告の提出、e-チケットの受領などさまざまな分野で存在する。そこで課題として、有意義な方法でイベントデータを活用することが挙げられる: 例えば洞察を提供し、ボトルネックを特定し、問題を予測し、ポリシー違反を記録し、対策を推奨し、プロセスを合理化する、などである。プロセスマイニングはまさにそれを行うことを目指している。

プロセスマイニングの出発点は**イベントログ**である。すべてのプロセスマイニング技術は、各イベントが**アクティビティ**(プロセスにおいて、明確に定義されたステップ) に対応付き、特定の**事例** (すなわち、プロセスのインスタンス) に関連するように、**順番に**記録できることを前提としている。また、イベントログがイベントに関する追加情報を持っている場合がある。実際、可能な限りプロセスマイニング技術では、例えばアクティビティを実行/開始する**リソース** (利用者やデバイス)、イベントの**タイムスタンプ**、またはイベントと一緒に記録されている**データ要素** (例えば注文量) などの情報を利用する。

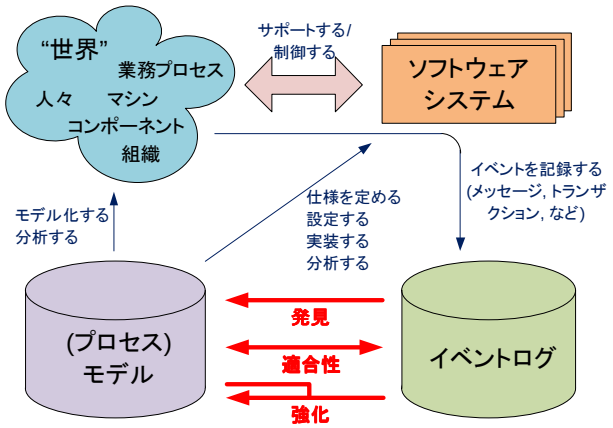

図 2. プロセスマイニングの 3 種類の主なポジショニング: (a) **発見**、(b) **適合性**検査、(c) **強化**。

図 2 に示すようにイベントログは 3 種類のプロセスマイニングを実施するのに利用できる。プロセスマイニングの最初のタイプは**発見** (discovery) である。発見手法はイベントログからモデルを前提の知識なしに生成する。プロセス発見は最もよく知られているプロセスマイニング技法である。多くの組織にとって、イベントログで示される実行例のみで本当のプロセスが実際に発見できることは驚くことである。プロセスマイニングの第二のタイプは**適合性** (conformance) であり、既存のプロセスモデルを同じプロセスのイベントログと比較する。適合性検査はログに記録されている現実がモデルと適合しているか、また逆も成り立つかを検査することができる。ここで留意すべきはいくつかのタイプのモデルが考えられることで、適合性検査を手続き型モデル、組織モデル、宣言的プロセスモデル、ビジネスルール/方針、法律、などに適用することができる。3 番目のタイプのプロセスマイニングは**強化** (enhancement) である。そのアイデアは、イベントログに記録された実際のプロセスに関する情報を使用して、既存のプロセスモデルを拡張したり改善することである。適合性検査がモデルと現実との間の合致度を評価するのに対し、この第三のタイプは既存のモデルを変更・拡張することを目指している。例えばイベントログのタイムスタンプを使うことで、ボトルネック、サービスレベル、スループット時間、頻度を表示するモデルへ拡張できる。

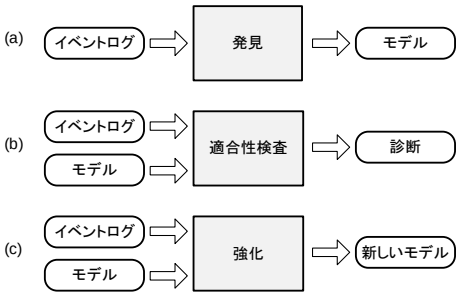

図 3. 入力と出力の観点からプロセスマイニングの三つの基本タイプを説明する: (a) 発見、(b) 適合性検査、(c) 強化。

図 3 を用い、入力と出力の観点からプロセスマイニングの三つのタイプを説明する。発見技術はイベントログを入力し、モデルを生成する。発見されたモデルは通常、プロセスモデル (例えばペトリネット、BPMN、EPC、UML アクティビティ図) であるが、モデルが別の観点を示している場合もある (例えばソーシャルネットワーク)。適合性検査技術は入力としてイベントログとモデルを用いる。出力結果は診断情報であり、モデルとログ間の差異と共通点を示す。モデルの強化 (修理または拡張) 技術も入力としてイベントログとモデルを用いる。出力結果は改良や拡張が施されたモデルである。

プロセスマイニングはさまざまな観点を扱う。**制御フローの観点**からは制御フロー、すなわちアクティビティの順序に焦点を当てる。この観点の目標は、可能なパスすべての良い特性を見つけることである。結果は通常、ペトリネットや他のプロセス表記法 (例えば、EPC、BPMN、UML アクティビティ図) で表現される。**組織の観点**からは、ログに隠れているリソースに関する情報に焦点を当てる。つまり、どのアクター (例えば、人、システム、ロール、部署) が関与し関連しているかである。この観点の目標は、ロールと組織単位で人々を分類することで組織を構成するか、ソーシャルネットワークを示すことである。**事例の観点**では事例の特性に焦点を当てる。もちろん事例はプロセス内のパスやそれを実行するアクターで特徴づけられる。しかし、事例は対応するデータ要素の値によっても特徴づけられる場合がある。例えば補充注文の場合、サプライヤーや注文数に興味があるかもしれない。**時間の観点**は、イベントのタイミングと頻度に関係する。イベントがタイムスタンプを持つ場合は、ボトルネック発見、サービスレベル測定、リソース使用率の監視、実行中の事例の残り処理時間の予測が行える。

マイニングの処理に関連してよく誤解される点がある。ベンダー、アナリスト、研究者の一部は、オフライン解析だけに使えるプロセス発見のためのデータマイニング技術をプロセスマイニングの範囲としている。これは真実では**ない**。そこで次の三つの特徴を強調する。

- **プロセスマイニングは制御フローの検出だけではない。**イベントログからプロセスモデルを発見することは、実務家と学者の両方にとって興味深い。そのため制御フローの発見はプロセスマイニングでもっとも刺激的であると見られている。しかしプロセスマイニングは制御フロー検出に限定されるものではない。プロセス発見はプロセスマイニングの三つの基本タイプ (発見、適合性、強化) の一つにすぎない。適用範囲は制御フローに限定されるものではなく、組織、事例、時間の観点も重要な役割を果たしている。
- **プロセスマイニングはデータマイニングの単なる一種ではない。**プロセスマイニングはデータマイニングと従来からあるモデル駆動型 BPM との間の “ミッシングリンク” として見ることができる。多くのデータマイニング技術はプロセス中心型ではない。並列を扱うプロセスモデルは決定木や相関ルールなどの単純なデータマイニング構造とは比べものにならない。そのため完全に新しいタイプの表現方法やアルゴリズムが必要である。
- **プロセスマイニングはオフライン解析に限定されるものではない。**プロセスマイニング技術は過去に発生したイベントデータから知識を抽出する。“事後” のデータを用いるものの、結果は実行中の事例に適用可能である。例えば、処理中の顧客注文の完了時間を、発見したプロセスモデルを用いて予測できる。

図 4 に示すビジネスプロセスマネジメント (BPM) のライフサイクルに、プロセスマイニングを当てはめてみよう。BPM のライフサイクルには、業務プロセスとそれに関連する情報システムの七

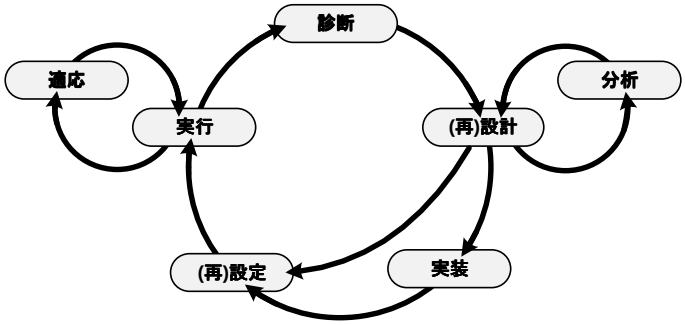

図 4. さまざまなフェーズを持つ BPM ライフサイクル; (実行段階を除き) プロセスマイニングはすべての段階で役に立つだろう。

つのフェーズがある。(**再**) **設計フェーズ**では、新規プロセスモデルを作成するか、既存のプロセスモデルを変更する。**分析フェーズ**では、候補のモデルとその代替案を分析する。(再) 設計フェーズの後、モデルを実装する (**実装フェーズ**) か、既存システムを (再) 設定する (**再設定フェーズ**)。**実行フェーズ**では設計したモデルを実行する。実行フェーズではプロセスが**監視**される。場合によってはプロセスを再設計せずに微調整を行う (**適応フェーズ**)。**診断フェーズ**では実行されたプロセスを分析し、結果によっては新たにプロセス再設計を行う。図 4 に示すように、プロセスマイニングはほぼすべてのフェーズで役立つ。もちろんプロセスマイニングは診断フェーズで役に立つが、診断フェーズに限定されるものではない。例えば実行フェーズでは、プロセスマイニングの手法を**運用サポート**に適用できる。履歴情報で学習したモデルを用いた予測と推奨を利用すると、実行中の事例に影響を与えられる。同様に意思決定支援は、プロセスの修正やプロセス (再) 設定の手助けに使える。

図 4 が BPM のライフサイクル全体を表しているのに対し、図 5 は具体的なプロセスマイニングの適用と成果物に焦点を当てており、プロセスマイニングプロジェクトにおける段階を表している。すべてのプロセスマイニングプロジェクトは、計画立案とその正当化 (ステージ 0) から始める。プロジェクトを立ち上げたら、イベントデータ、モデル、目的、疑問点を、システム、ドメインの専門家、および管理部門から抽出する (ステージ 1)。このためには図 5 に示すように、利用可能なデータを理解し (“分析に何が使えるか?”)、ドメインを理解して (“重要な疑問点は何か?”)、成果物 (つまり、履歴データ、手製のモデル、目的、疑問点) にまとめることが求められる。ステージ 2 では、制御フローモデルを構成し、イベントログを関係づける。ここでは自動プロセス発見技術が使える。発見されたプロセスモデルからいくつかの疑問点に対する答えがわかり、再設計や適応行動のきっかけになるかもしれない。さらに、モデルを用いたイベントログの分離や選択が可能である (例えば稀な活動や異常値の事例を除いたり、不足しているイベントを挿入する)。場合によっては、同一のプロセスインスタンスに属するイベントを関連付けるのが大変なことがある。残りのイベントはプロセスモデルのエンティティに関連している。プロセスが比較的構造化されている場合、ステージ 3 では他の観点 (例えば、データ、時間、リソース) を用いて制御フローモデルを拡張できる。ステージ 2 で確立されたイベントログとモデルとの関係はモデルを拡張するのに使える (例えばイベントのタイムスタンプを用いてアクティビティの待ち時間を推定するなど)。これは追加の疑問点に対する答えであり、新たな行動のきっかけになるかもしれない。最後に、ステージ 3 で構築したモデルは運用サポート (ステージ 4) に使うことができる。過去のイベントデータから抽出した知識を実行中の事例の情報と結合すると、調停、予測、推奨が行える。ステージ 3 と 4 に到達できるのは、プロセスが十分に安定して構造化されている場合のみである。

図 5 に示すように既にすべての段階をサポートできる技術やツールがある。しかしプロセスマイニングは比較的新しいパラダイムであり、現在入手可能なツールのほとんどは依然として未熟である。また、将来の利用者がプロセスマイニングの潜在能力と限界に気づかないかもしれない。そこで、本マニフェストでは最先端の技術に興味を持つプロセスマイニングの利用者、研究者、開発者向けに、指針 (3 節) と課題 (チャレンジ; 4 節) の一覧を示す。

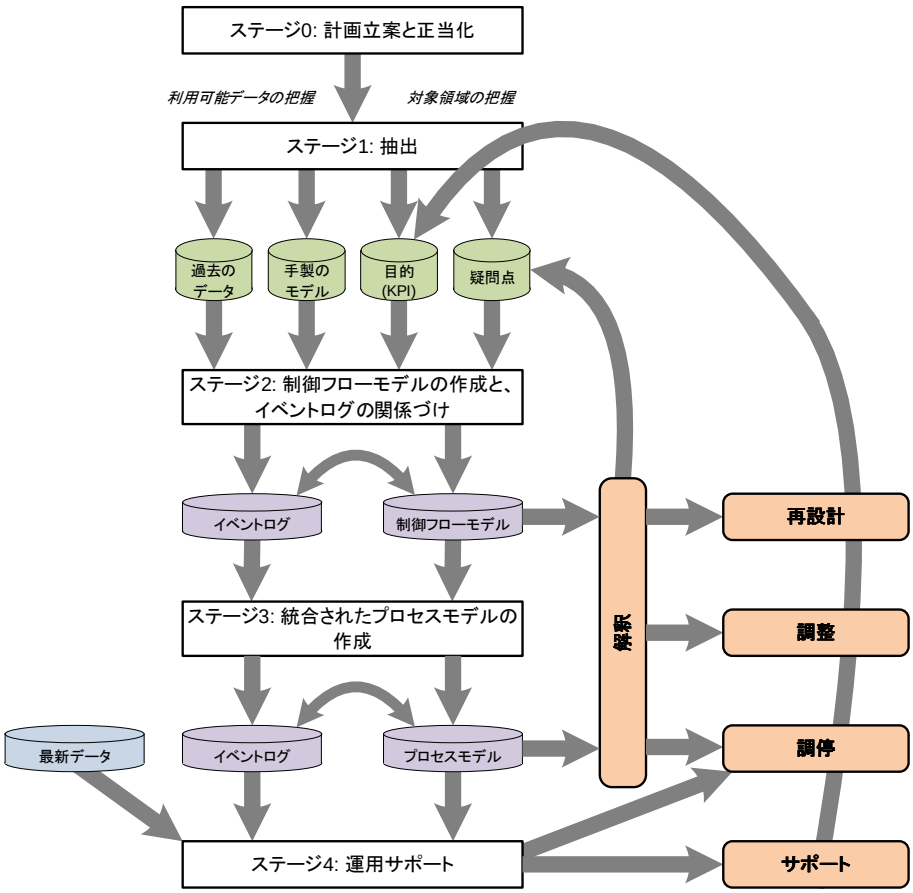

図 5. 五つのステージから成るプロセスマイニングプロジェクトを記述する L^* ライフサイクルモデル: 計画立案と正当化 (ステージ 0)、抽出 (ステージ 1)、制御フローモデルの生成とイベントログの関係づけ (ステージ 2)、統合プロセスモデルの作成 (ステージ 3)、運用サポート提供 (ステージ 4)。

3 指針

あらゆる新技術と同様に、プロセスマイニングを現実世界に適用する際に起こる、分かりきった間違いがある。そこで、利用者/分析者のそのような間違いを防ぐために、以下に 6 個の**指針**を示す。

3.1 GP1: イベントデータは第一級市民として扱われるべきだ

プロセスマイニングの出発点は、記録されたイベントである。イベントの集合を**イベントログ**と言っているが、必ずしもイベントを専用のログファイルに格納する必要はない。イベントは、データベースのテーブル、メッセージログ、メールアーカイブ、トランザクションログ、その他のデータソースに格納される。格納形式よりも重要なのは、イベントログの**品質**である。プロセスマイニングの結果の質は大いに入力に依存する。したがって、分析対象のプロセスを実行する情報システムにおいて、イベントログは**第一級市民**として扱われるべきである。残念なことに、イベントログは大抵デバッグやプロファイリングに使うための単なる “副産物” である。例えば、Philips Healthcare の医療機器ではイベントを記録しているが、それは単にソフトウェア開発者がソースコードに “print 文” を挿入していたという理由しかない。

ソースコードにこのようなステートメントを追加するためのいくつかの非公式のガイドラインがあるが、イベントログの質を向上させるためにはより体系的なアプローチが必要である。イベントデータは (二流市民ではなく) 第一級市民と考えるべきである。

イベントデータの品質を判断するいくつかの基準がある。イベントは**信頼できる**べきだ。つまり、記録されたイベントは実際に発生したことであり、イベントの属性は正確だと見なせるべきである。

イベントログは**完全である**、つまり、ある範囲においてイベントには欠落がないべきである。記録されたイベントはすべて明確に定義された**セマンティクス**を持つべきである。さらに、イベントを記録する際には、プライバシーやセキュリティの観点からイベントデータが**安全**であるべきだ。例えば、記録されているイベントの種類とそれらがどのように使われたかを、アクターは承知しておくべきである。

表 1. イベントログの成熟度レベル。

レベル	特性
⋆⋆⋆⋆⋆	最高レベル: イベントログの品質が優れていて (すなわち、信頼でき完全である)、イベントが明確に定義されている。イベントは自動的に、体系的に、確実に、安全な方法で記録される。プライバシーとセキュリティが適切に考慮されている。さらに、記録されたイベント (と属性のすべて) は、明確なセマンティクスを持っている。これは 1 個以上のオントロジーが存在することを意味する。イベントおよびその属性はこのオントロジーに対応する。 **例**: BPM システムの意味的注釈付きのログ。
⋆⋆⋆⋆	イベントが自動的に、体系的かつ信頼できる方法で記録される。すなわち、ログは信頼でき完全である。⋆⋆⋆ レベルのシステムとは異なり、このようなプロセスのインスタンス (事例) やアクティビティなどの概念が明示的にサポートされている。 **例**: 従来の BPM/ワークフローシステムのイベントログ。
⋆⋆⋆	イベントは自動的に記録されるが、体系的な取り組みになっていない。しかし、⋆⋆ レベルのログとは異なり、イベントが現実を反映して記録されていることがある程度保証されている (すなわち、イベントログは信頼できるが、必ずしも完全ではない)。例えば、ERP システムで記録されたイベントを考えてみると、イベントは、さまざまなテーブルから抽出する必要があるが、情報は正しいと仮定できる (例えば、ERP で記録された支払いは実際に存在し、その逆も真と見なせる)。 **例**: ERP システム内のテーブル、CRM システムのイベントログ、メッセージングシステムのトランザクションログ、先端技術システムのイベントログなど
⋆⋆	イベントは情報システムの副産物として自動的に記録される。対象範囲はまちまちである、すなわち、どのイベントを記録するかを決定する体系的な取り組みが存在しない。また情報システムをバイパスすることが可能である。したがってイベントが存在しなかったり、正しく記録されていないことがある。 **例**: 文書や製品管理システムのイベントログ、組み込みシステムのエラーログ、サービスエンジニアのワークシート、等。
⋆	最下位レベル: イベントログの品質が悪い。記録されたイベントは、現実に対応していない場合があり、イベントが失われていることがある。手作業で記録されるイベントログは通常そのような特性を持つ。 **例**: 組織を経由しながら紙に残された痕跡 (“付箋”)、紙ベースの医療記録など。

表 1 に、イベントログに関する優れた品質 (⋆⋆⋆⋆⋆) から低品質 (⋆) までの 5 段階の成熟度の定義を示す。例えば、Philips Healthcare のイベントログは、⋆⋆⋆ レベルに相当する、すなわちイベントは自動的に記録され、記録された挙動は現実と一致するが、イベントにセマンティクスを割り当て、カバレッジが特定のレベルに保証されるための体系的な取り組みがない。プロセスマイニング技術は、⋆⋆⋆⋆⋆ レベル、⋆⋆⋆⋆ レベル、⋆⋆⋆ レベルのログに適用可能である。原理上は ⋆⋆ レベルや ⋆ レベルのイベントログにも適用可能である。しかし、そのようなログを用いた分析には問題が多く結果を信頼できない。事実、⋆ レベルのログにプロセスマイニングを適用するのは意味がない。

プロセスマイニングを役立てるには、最高品質のイベントログを目指すべきである。

3.2 GP2: ログの抽出は、疑問点に応じて行うべきだ

図 5 に示すように、プロセスマイニングは疑問点に応じて行う必要がある (疑問点駆動型)。具体的な疑問点がなければ、意味あるイベントデータを抽出するのは非常に困難である。例えば SAP などの ERP システムのデータベース内にある何千ものテーブルを考えてみると、具体的な疑問点がなければデータ抽出に関連するテーブルを選択することは不可能である。

図 1 に示したようなプロセスモデルは、ある一つのタイプの事例 (すなわち、プロセスインスタンス) のライフサイクルを述べている。したがって、プロセスマイニング技法を適用する前に、分析する事例のタイプを選択する必要がある。この選択は答えが必要な疑問点を用いて決めるべきであるが、これは単純ではないかもしれない。例えば顧客注文処理を考えてみると、顧客は一回の発注で複数個の製品を発注する場合があるため、複数の明細行を持つ注文伝票になるかもしれない。そのため注文伝票 1 件で複数の配送が発生することがある。また、一つの配送伝票が複数の注文伝票の明細に対応する場合がある。したがって、注文伝票と配送伝票は多対多の関係であり、注文伝票と明細行は一対多の関係である。注文、明細行、配送に関連するイベントデータを持つデータベースが与えられると、さまざまなプロセスモデルを発見できる。まず、個々の注文のライフサイクルを記述することを目的として、データを抽出できる。また一方、個々の注文明細行のライフサイクルや個々の配送のライフサイクルを発見することを目的として、データを抽出することも可能である。

3.3 GP3: 並列、選択、その他の基本的な制御フロー構成をサポートすべきだ

プロセスモデリング言語はたくさん存在する (例えば、BPMN、EPC、ペトリネット、BPEL、UML アクティビティ図)。これらの言語の中には多種のモデリング要素を用意しているものがある (例えば BPMN では 50 以上の異なるグラフィカルな要素がある) のに対し、他の言語は基本的な要素だけを用意している (ペトリネットが用意している要素は 3 種類だけである: プレース、トランジション、アーク)。制御フローの記述は、すべてのプロセスモデルにおける重要要素である。すべての主流の言語で用いられている基本的なワークフロー構成要素 (**パターン**として知られている) は、シーケンス、並列ルーティング (AND-split/join)、選択 (XOR-split/join)、およびループである。もちろん、これらのパターンはプロセスマイニング技術でサポートされるべきである。しかし、いくつかの技術は並列性を扱えず、マルコフ連鎖/推移システムのみをサポートしている。

図 6 に、並列性を発見できない (AND-split/join を持たない) プロセスマイニング技術を使用したときの結果を示す。イベントログ $L = \{\langle A, B, C, D, E\rangle, \langle A, B, D, C, E\rangle, \langle A, C, B, D, E\rangle, \langle A, C, D, B, E\rangle, \langle A, D, B, C, E\rangle, \langle A, D, C, B, E\rangle\}$ があるとする。L は A から始まり E で終わる事例を含む。B、C、D は A と E の間で任意の順番で発生する。図 6(a) に示した BPMN モデルではプロセスを二つの AND ゲートウェイを用いてコンパクトに表現している。プロセスマイニング技術が AND ゲートウェイをサポートしていないとしよう。この場合、図 6 に示した残り二つの BPMN モデルが候補として考えられる。図 6(b) の BPMN モデルはコンパクトであるが余計な挙動も許してしまう (例えば、$\langle A, B, B, B, E\rangle$ のような事例はイベントログには存在しないがモデルでは可能である)。図 6(c) の BPMN モデルは L に出現するすべての事例を表現しているが、すべてのシーケンスを直接的に表現しているため、コンパクトなログの表現ではない。この例は、並列に動作するアクティビティが多く存在する現実のモデルに対して、並列がサポートされていないと得られるモデルがあまりにもアンダーフィット (余計な挙動を許しすぎる) か複雑すぎることを示している。図 6 に示したとおり、基本的なワークフローパターンをサポートすることは重要である。上記の基本パターンに加え、OR-split/join もサポートするのが望ましい。これにより包含的判断 (inclusive decision) と部分的な同期化をコンパクトに表現できるようになる。

3.4 GP4: イベントはモデル要素に関連しているべきだ

2 節に示すように、プロセスマイニングが制御フロー発見に限定されるというのは間違った考えである。図 1 に示すように、発見されたプロセスモデルはさまざまな観点 (組織の観点、時間の観点、データの観点、等) を持つ。また図 3 に示すように、発見はプロセスマイニングの三つのタイプの一つにすぎない。その他二つのタイプのプロセスマイニング (適合性検査と強化) は**モデル内の要素**と**ログ内のイベント**間の関係に強く依存する。この関係は、モデル上でイベントログを “再生” するのにも用いられる。再生することで、イベントログとモデルの間の不一致 (例えば、モデルに従うとログ内のいくつかのイベントがあり得ない) が明らかになるかもしれない。適合性検査技術はこのよう

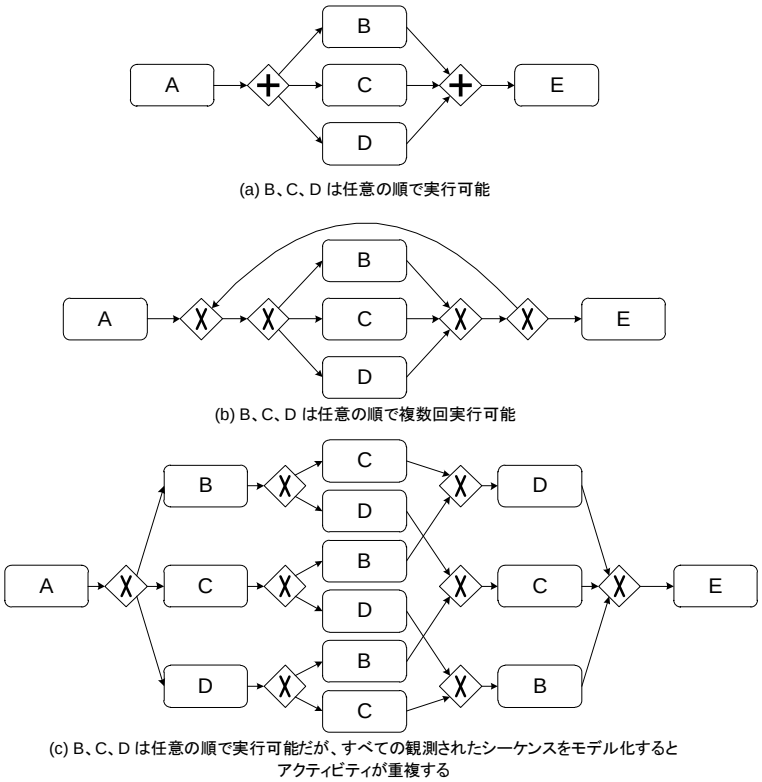

図 6. 並列 (AND-split/join) を直接的に扱えない場合の問題の例。この例では 3 個のアクティビティ(B, C, D) のみが並列である。もし 10 個の並列なアクティビティがあるときのプロセスモデルを想像してみよう ($2^{10} = 1024$ 状態と、$10! = 3,628,800$ 通りの実行シーケンスが存在する)。

な不一致を定量化し診断する。再生時に、イベントログのタイムスタンプは時間的な挙動の解析に使える。因果関係にあるアクティビティ間の時間差が分かると、モデルで予想待ち時間を扱えるようになる。これらの例に示すように、ログ内のイベントやモデルの要素間の関係が分かると異なるタイプの分析が可能となる。

場合によっては、そのような関係を確立するのが非常に困難である。例えば、一つのイベントが二つの異なるアクティビティに対応したり、どのアクティビティに対応するかがはっきりしない場合がある。プロセスマイニングの結果を適切に解釈するためは、このようなあいまいさを除く必要がある。イベントとアクティビティを対応づける問題の他に、イベントをプロセスインスタンスに関連付けるときの問題がある。これは普通、**イベントの相関**と呼ばれる。

3.5 GP5: 現実の意図的な抽象化としてモデルを扱うべきだ

イベントデータから導出されたモデルは、**現実のビュー**を提供する。このようなビューは、挙動を意図的に抽象化したものを提供すべきである (挙動はイベントログに記録されている)。与えられたイベントログに対して、便利なビューが複数あるかもしれない。また、さまざまな関係者がさまざまなビューを必要とする場合がある。実際、イベントログから発見されたモデルを “マップ”(地図など) であると考えるべきである。以下では重要な洞察を二つ示す。

まず第一に、特定の地域のための “唯一のマップ” なるものは存在せず、用途ごとに異なるマップがある: 道路地図、ハイキングマップ、サイクリングマップなど。これらのマップはすべて同じ現実のビューを表しており、“完璧なマップ” の存在を仮定するのは無理がある。同じことがプロセスモデルにも当てはまる: モデルは特定のタイプの利用者にとって意味のある事柄を強調すべきである。そのためには発見されたモデルを異なる観点 (制御フロー、データフロー、時間、リソース、コスト、等) に関して、異なる粒度と精度で表示する。例えば経営者はコストについて要約版のプロセスモデルで見たいだろうし、プロセスアナリストは通常フローとの違いが分かりやすい詳細プロセ

スモデルを見たがる。また、関係者ごとに異なるレベルでプロセスを活用する: **戦略レベル** (このレベルでの意思決定は長期的な効果を持ち、長期間にわたるイベントデータに基づき決められる)、**戦術レベル** (このレベルでの意思決定は中期的な効果を持ち、最新のデータに基づき決められる)、**運用レベル** (このレベルの意思決定は即効性で、実行中の事例のイベントデータに基づき決められる)。

第二に、理解しやすいマップを生成するなら、地図作成からアイデアを借りるのがよい。例えば、道路地図における重要性の低い道路や都市の抽象化である。重要性の低いものは削除されたりひとまとめにされる (例えば街路や郊外が一緒になって都市として表記される)。地図製作者は重要でない詳細を排除するだけでなく、重要な特徴を色で強調する。また、グラフィカルな要素は重要度を示すために特有の大きさを持つ (例えば、線や点の大きさが異なる)。地図の x 軸と y 軸は明確な意味を持つ (地図のレイアウトはいい加減ではなく、要素の座標には意味がある)。これに比べて、現在主流のプロセスモデルでは色、大きさ、位置の特徴を用いておらず、モデルを分かりやすくしようとしていない。しかし地図作成から得られるアイデアは、プロセスマップを作るときに簡単に組み込むことができる。例えばアクティビティの大きさは、実行回数や他の重要性を示す特性 (コストやリソースの使用) を示すのに使える。矢印の幅は因果関係の強さを示すことができ、色はボトルネックを強調するのに使える。

上記の観察は、正しい表現を選択し利用者向けに微調整することが重要であることを示しており、エンドユーザ向けに結果を可視化し、適切なモデルになるよう発見アルゴリズムを導くときに大切なことである (課題 C5 も参照)。

3.6 GP6: プロセスマイニングは継続的なプロセスであるべきだ

プロセスマイニングを用いると、直接イベントデータとつながっている、有意義な “マップ” を提供することができる。過去のイベントデータと現在のデータの両方をこのようなモデル上に投影することができる。また、プロセスは分析している間にも変化する。プロセスの動的な性質を考えると、プロセスマイニングを一回だけの作業と見なすのはよくない。そのため、一つの固定されたモデルを作成することではなく、生き生きとしたプロセスモデルを作ることで利用者やアナリストが日課として見るようになることが目標である。

ジオタグを用いたマッシュアップの使い方と比較してみると、Google マップを使ったマッシュアップ (例えば、交通状況、不動産、ファーストフードレストラン、映画の上映時間などの情報を地図上に投影するアプリケーション) が数多く存在する。そのようなマップを使用してシームレスにズームイン、ズームアウトしながらアプリケーションと対話することができる (例えば、交通渋滞をマップ上に投影し、特定の問題を選択して詳細を見る)。また、リアルタイムのイベントデータを用いてプロセスマイニングを行えるはずである。“マップのメタファー” を使うと GPS 座標情報を持つイベントがあればリアルタイムで地図上に表示できる。カーナビゲーションシステムのように、プロセスマイニングツールは次のようにエンドユーザを支援できる: (a) プロセス内をナビゲートする、(b) プロセスマップに動的な情報を投影する (例えば、業務プロセスの “交通渋滞” を示す)、(c) 実行中の事例に関する予測を提供する (例えば、遅延している事例の “到着時間” を推定する)。これらの例で分かるように、プロセスモデルを積極的に使用しないのは宝の持ち腐れである。したがって、プロセスマイニングは、さまざまな時間スケール (分、時間、日、週、月) に応じて、すぐに実施可能な情報を提供する継続的なプロセスと考えるべきである。

4 課題

プロセスマイニングは、重要な運用プロセスを管理する必要がある近代的な組織のための、重要なツールである。一方ではイベントデータが急速に増加してきており、他方では、コンプライアンス、効率性、顧客サービスに関連する要件を満たすために、プロセスと情報が完全に連携する必要がある。プロセスマイニングの実用性にもかかわらず、取り組むべき重要な課題 (チャレンジ) がまだある。このことはプロセスマイニングが新興の専門分野であることを表している。以下に課題のいくつかを示す。このリストは完全なものではないし、将来的に新たな課題が出現するか、プロセスマイニングの進歩によって消えるだろう。

4.1 C1: イベントデータの検索、マージ、クリーニング

プロセスマイニングに適したイベントデータを抽出するのは今でも相当な手間を要することで、いくつかのハードルを克服する必要がある:

– データがさまざまな情報源に**分散している**場合がある。そのため情報をマージする必要があるが、異なるデータソースで異なる識別子が使用されていると、難しい問題になりがちである。例えば、個人を識別するのにあるシステムは名前と生年月日を用いており、別のシステムは社会保障番号を用いているような場合である。
– イベントデータは多くの場合 “オブジェクト中心” であり、“プロセス中心” ではない。例えば、個々の製品、パレット、およびコンテナは RFID タグを持っており、記録されたイベントはこれらのタグに対応する。しかし、特定の顧客注文を監視するには、オブジェクト中心のイベントだと、マージして前処理する必要がある。
– イベントデータは**不完全である**場合がある。よくある問題は、イベントが明示的にプロセスインスタンスに対応しないことである。この情報を得ることは可能な場合が多いが、かなりの手間を要する場合がある。また、いくつかのイベントで時刻情報が欠落している場合がある。利用可能な時刻情報を活用するために、欠落したタイムスタンプの補間が必要になるかもしれない。
– イベントログには、**異常値** (例外的な挙動 (**ノイズ**)) が含まれる場合がある。異常値を定義するには? このような異常値を検出する方法は? これらの疑問点に答えるにはイベントデータをきれいにする必要がある。
– ログは**粒度が異なるレベル**のイベントを含んでいる場合がある。病院情報システムのイベントログには、イベントとして簡単な血液検査や、複雑な外科手術が含まれている。タイムスタンプも異なる精度が混在し、ミリ秒精度 (2011 年 9 月 28 日 11 時 28 分 32.342 秒) や粗い日付情報 (2011 年 9 月 28 日) などがある。
– イベントは特定の**コンテキスト** (天候、作業負荷、曜日、等) で発生する。このコンテキストは特定の現象などを説明するかもしれない (例えば、応答時間が通常より長くその原因が進行中の作業や休日であるため、など)。分析のためには、このコンテキストを一緒に利用することが望ましい。これは、コンテキストデータとイベントデータのマージを意味する。しかしあまり多くの変数を追加すると “次元の呪い” によって分析が困難になる。

上記の問題に対処するためには、より良いツールと方法論が必要である。また前述のように、イベントログを単なる副産物ではなく第一級市民として扱う必要がある。目標は ⋆⋆⋆⋆⋆ レベルのイベントログ (表 1 を参照) を取得することである。このとき、データウェアハウスで学んだ教訓が、高品質のイベントログを確保するために役立つ。例えば、データ入力時に簡単なチェックをすることで、不正確なイベントデータの割合を劇的に減らすことができる。

4.2 C2: 多様な特性を有する複雑なイベントログを扱う

イベントログが非常に異なる特性を持つ場合がある。いくつかのイベントログは非常に大量であるため取り扱いが困難であるのに対し、他のイベントログは小規模すぎて信頼できる結論を得られないことがある。

いくつかの分野では、驚くほど大量のイベントが記録される。したがってパフォーマンスとスケーラビリティを向上させる努力が必要になる。例えば、ASML (訳注: オランダに本社を置く半導体製造用露光装置メーカー) はウェハースキャナすべてを継続的に監視している。これらのウェハースキャナは、さまざまな組織 (Samsung、Texas Instruments など) に採用されており、チップ製造に用いられている (チップの約 70 %が ASML のウェハースキャナで製造されている)。既存ツールはそのようなドメインで収集されたペタバイトに及ぶデータは既存ツールでは扱いきれない。イベントの記録量に加え、事例あたりの平均イベント数、事例間の類似性、ユニークなイベント数、ユニークなパス数などの特徴も存在する。イベントログ $L1$ が次の特徴を持つ場合を考えてみよう: 1000 事例、事例あたり 10 イベント、変動がほとんどない (つまり事例が、ほとんど同じか類似したパスになっている)。もう一つのイベントログ $L2$ は 100 事例のみだが事例あたり 100 イベント含まれていてすべての事例がユニークなパスである。ログ量はどちらのログも同程度 (約 1 万イベント) であるが、明らかに $L1$ より $L2$ を分析するほうが難しい。

イベントログは一部分の挙動しか含まないので、イベントログが完全と見なしてはいけない。プロセスマイニング技術は “開世界仮説 (Open world assumption)” (何かが発生しなかったという事実があっても、それを実現できないというわけではない) を用いて不完全性に対処する必要がある。ただしこの前提により、多くのバリエーションを持つ少量のイベントログの扱いが難しくなる。

前述のように、ログには抽象化レベルのとても低いイベントが含まれているものがある。このようなログは非常に大量になる傾向があるが、低レベルのイベントそれぞれには関係者は興味がない。

そのため低レベルのイベントを集約して、高レベルのイベントにしたい。例えば、ある種の患者グループに対する診断と治療プロセスを分析したいとき、病院の検査室にある情報システムに記録されている個々のテスト記録には関心がないだろう。

現段階では、あるイベントログがプロセスマイニングに適しているかを調べるために試行錯誤する必要がある。そのため、あるデータセットに対する実行可能性調査を迅速に行えるツールが必要である。そのようなテストにより潜在的なパフォーマンスの問題が分かるはずで、ログが完全にはほど遠いか詳細すぎる場合は警告する必要がある。

4.3 C3: 代表的なベンチマークを作成する

プロセスマイニングは新興技術である。このため良いベンチマークがまだ不足している。例えば、数十のプロセス発見技術が利用可能であり、いくつものベンダーが製品を提供しているが、技術の質について意見の一致は得られていない。機能性やパフォーマンスには大きな違いがあるが、さまざまな技術やツールを比較することは困難である。そのため、データセットと代表的な品質基準から構成される、良いベンチマークを開発する必要がある。

古典的なデータマイニング手法については多くの良いベンチマークが用意されている。これらのベンチマークはツール提供者や研究者に刺激を与え、技術パフォーマンスが向上した。プロセスマイニングの場合、ベンチマークの開発がより困難である。例えば、1969 年に Codd が考え出したリレーショナルモデルは単純であり、広く支持されている。その結果、あるデータベースから別のデータベース用にデータを変換するのは手間ではなく、解釈の問題も発生しない。プロセスにはそのような単純なモデルが存在しない。プロセスモデリング向けに提案された標準は大変複雑であり、完全に準拠した機能をサポートするベンダーはほとんどない。プロセスは表形式のデータよりも圧倒的に複雑なのである。

それでもなお、プロセスマイニング向けの代表的なベンチマークを作成することは重要である。いくつかの初期のベンチマークは既に利用可能である。例えばプロセスマイニングの結果の品質を測定するためのさまざまなメトリックがある (フィットネス、単純さ、精度、汎化)。また、いくつかのイベントログは公開されている (www.processmining.org)。例えば、タスクフォース主催の第 1 回ビジネスプロセスインテリジェンスチャレンジ (BPIC'11) ではそのイベントログが使用された (doi:10.4121/uuid:d9769f3d-0ab0-4fb8-803b-0d1120ffcf54)。

現実のデータセットに基づいたベンチマークが必要であるが、特定の特性を備えた人工のデータセットを作ることも必要である。そのような合成データセットは不完全なイベントログ、ノイズの多いイベントログ、特定のプロセスの集合向けに、プロセスマイニング技術を開発するのに役立つ。

代表的なベンチマークの作成に加えて、プロセスマイニングの結果の品質を判断する基準について合意が必要である (課題 C6 も参照)。加えて、データマイニングにおける**クロスバリデーション**法が結果を判断するために使える。例えば、k-分割検査では、イベントログを k 個に分割し、$k-1$ 個の分割を用いてプロセスモデルを学習し、残りの分割について適合性検査技術を用いて結果を判断することができる。これを k 回繰り返すことで、モデルの品質について洞察を得る。

4.4 C4: コンセプトドリフトの扱い

コンセプトドリフト *(concept drift)* は、分析中にプロセスが変化している状況を意味する。例えば、当初のイベントログでは 2 個のアクティビティが並列だったのに対し、後ではそれらのアクティビティが直列になっている、というようなことである。周期/季節変化 (例えば、"12 月に需要が増加する" や "金曜日の午後には作業可能な従業員が少なくなる") や状況の変化 (例えば、"市場で競争が激しくなってきた") によってプロセスは変化する。このような変化はプロセスに影響を与えるため、それらを検出・分析することが重要である。プロセスのコンセプトドリフトを見つけるには、イベントログを小さなログに分割し、小さなログの "足跡" を分析する。このような "二次" 分析では、より多くのイベントデータを必要とする。しかし、定常状態にあるプロセスは滅多になく、コンセプトドリフトを理解することはプロセス管理にとってもっとも重要なことである。そのため、的確にコンセプトドリフトを分析するための、更なる研究とツール支援が必要である。

4.5 C5: プロセス発見に使用される表現バイアスを改善する

プロセス発見技術は特定の言語 (例えば BPMN やペトリネット) を使用してモデルを生成する。しかし、結果の可視化と、実際に発見を行う時に用いられる表現とを分離することは重要である。ター

ゲット言語の選択は多くの場合に暗黙の仮定を含むことになり、探索空間を制限する。つまり、ターゲット言語で表現できないプロセスは発見できない。発見プロセス中に使われる、いわゆる “表現バイアス (representational bias)” は意識的な選択であるべきで、好みのグラフィカルな表現に左右されてはいけない。

例えば図 6 を見てみると、ターゲット言語が並列を表現できるか否かは、発見したモデルの可視化とアルゴリズムで検討したモデルのクラスの両方に影響を与える。表現バイアスが並列を表現できず (図 6(a) が無理)、同じラベルを持つ複数のアクティビティが存在できない (図 6(c) が無理) 場合、図 6(b) に示したような問題の多いモデルのみが可能となる。この例は、表現バイアスを選択するときに、より慎重によく検討して選択するべきであることを示している。

4.6 C6: フィットネス、単純さ、精度、汎化など品質基準間のバランス

イベントログは完全には程遠いものが多い、つまり一部分の挙動のみが与えられる。プロセスモデルは通常、指数関数的または無限に長い、さまざまなトレース (ループがある場合) を許容する。また、一部のトレースが他よりはるかに低い発生確率を持つことがある。したがって、すべての起こりえるトレースがイベントログに存在すると考えるのは非現実的である。完全なログを想定するのが現実的でないことを示すために、10 個のアクティビティを含みそれらが並列に実行可能なプロセスと、それに対応する 10,000 件のログが存在する場合を考えてみると、10 個の並列に動作するアクティビティを持つモデルにおける可能な並び方は 10！= 3,628,800 通りになる。したがって、潜在的なトレース (3,628,800) よりもログが少なく (10,000)、すべての並び方がログに存在するということはあり得ない。ログに何百万もの事例がある場合でも、すべての可能なバリエーションが存在することは滅多にない。加えて複雑になるのは、いくつかの場合が他より発生しにくいということである。これらは、“ノイズ” と考えることができる。そのようなノイズが多い挙動に対して合理的なモデルを構築することは不可能である。そのため、適合性検査を使用して発生しにくい挙動を調査することが望ましい。

ノイズと不完全性が存在するとプロセス発見が難しい問題になる。事実、四つの競合する品質の次元がある: (a) フィットネス、(b) 単純さ、(c) 精度、(d) 汎化。良い**フィットネス**を持つモデルは、イベントログに含まれる挙動のほとんどを表せる。ログ内のすべてのトレースがすべてモデルによって再生できる場合、モデルには完璧なフィットネスが備わっている。ログに含まれる挙動を説明できる**最も単純な**モデルは最高のモデルである。この原理はオッカムの剃刀として知られている。フィットネスと単純さだけでは、発見されたプロセスモデルの品質を判断するには十分ではない。例えば、イベントログ内のすべてのトレースを再生可能な非常に単純なペトリネット (“フラワーモデル”) を構築するのはとても簡単である (ただし同じアクティビティの集合に対応する他のイベントログも含んでしまう)。(訳注: フラワーモデルは一つのプレースを中心として、それとトランジションとを往復するアークからなる組が複数個取り囲んでいる形状のため、花のように見えるグラフである。) 同様に、イベントログに含まれる挙動だけを持つモデルは望ましくない。前述のようにログは一部分の挙動のみからなり、起こりえるトレースの多くはまだ発生していない。“あまりにも多く” の挙動は許容しないモデルは**正確**である。明らかに “フラワーモデル” は精度を欠いている。精度の低いモデルは “アンダーフィッティング” である。アンダーフィッティングはログ中の一部分の挙動をモデルが汎化しすぎるという問題である (すなわち、ログに含まれるものと大きく異なる挙動をモデルが許容してしまう)。モデルは動作を汎化すべきだが、ログに見られる一部分の例によって動作を制限すべきではない。**汎化され**ていないモデルは、“オーバーフィッティング” である。オーバーフィッティングは、ログが一部分の挙動を保持しているのに過ぎないのに、あまりにも特定のモデルが生成される問題である (すなわち、モデルは特定のサンプルのログを説明するが、同じプロセスの別のサンプルログに対して、全く異なるプロセスモデルを生成する)。

フィットネス、単純さ、精度、汎化のバランスを取ることは難しい。強力なプロセス発見技術がさまざまなパラメータを提供するのはこのためである。4 種類の競合する品質次元をうまく組み合わせた、よりよいアルゴリズムが必要である。また、用いられたすべてのパラメータをエンドユーザが理解できるようにすべきである。

4.7 C7: 組織横断的なマイニング

従来、プロセスマイニングは、単一の組織内で適用されている。しかし、サービス技術、サプライチェーンの統合、クラウドコンピューティングが普及するにつれて、複数の組織のイベントログを用いた分析が可能となる。原則として、**組織横断的なプロセスマイニング**には 2 種類のタイプがある。

第一に、プロセスインスタンスを処理するために異なる組織が協力し合うような協働環境を考察してみよう。そのような組織横断的なプロセスは “ジグソーパズル” と考えることができる。すなわち全体的なプロセスを部分に分割し、事例の実行に協力しあう組織間に分散させたものである。関与する組織の一つのイベントログを分析するだけでは不十分である。隅から隅までのプロセスを発見するには、異なる組織のイベントログをマージする必要がある。組織の境界を越えてイベントを関係づけるので簡単な作業ではない。

第二に、経験、知識、共通のインフラストラクチャを共有しながら、異なる組織が本質的に同じプロセスを実行する環境を考察してみよう。例えば Salesforce.com では、多くの組織の販売プロセスが Salesforce によって管理・サポートされている。これらの組織は、インフラ (プロセス、データベース、など) を共有する一方、同じプロセスの修正版を使えるようにシステムを設定できるため、厳格なプロセスモデルに従わなくてよい。別の例として、自治体内で実行される基本的なプロセス (例えば、建築許可の発行) では、国内のすべての自治体は同一の基本的なプロセス群をサポートする必要があるが、違いもあるかもしれない。もちろん、異なる組織間での差異を解析することは興味深い。組織横断的プロセスマイニングの結果を用いて、これらの組織がお互いから学びあい、サービスプロバイダーがサービスを改良し、付加価値サービスを提供することができる。

両方のタイプの組織横断的プロセスマイニング向けに、新しい分析技術を開発する必要がある。これらの技術はまた、プライバシーやセキュリティ問題を考慮すべきである。競争上の理由や信頼の欠如のため、組織が情報を共有したくない場合がある。このためプライバシーを保護できるプロセスマイニング技術の開発が重要である。

4.8 C8: 運用サポートの提供

当初、プロセスマイニングは履歴データの分析に焦点を当てていた。しかし今日では、イベントが発生すると多くのデータソースが (ニア) リアルタイムで更新され、またイベントを分析するのに充分な計算パワーが存在する環境となった。したがって、プロセスマイニングをオフライン分析に限定せずに、オンラインの運用サポートにも使用するべきである。3 種類の運用サポート活動が存在する: **検出**、**予測**、**推奨**。事前に定義されたプロセスから事例が逸脱した瞬間に検出し、システムがアラートを生成することができる。オフラインのやり方ではなく、即時に (まだ物事に影響を与えられるうちに) 通知を生成したい場合が多くある。履歴データは予測モデルを構築するために使用することができる。これらを用いて、実行中のプロセスインスタンスを適切に制御できる。例えば、ある事例の残り処理時間を予測することが可能である。このような予測に基づいて、推奨システムを構築し、コストを削減したり、フローの時間を短縮するための行動を提案できる。このようなオンライン環境でプロセスマイニングの手法を適用するには、計算パワーとデータ品質の面で新たな課題が発生する。

4.9 C9: 他のタイプの分析と組み合わせたプロセスマイニング

オペレーション管理、とくにオペレーションリサーチは、モデルに大きく依存する管理科学の一分野である。線形計画法、プロジェクトプランニングからキューイングモデル、マルコフ連鎖、シミュレーションに至るまでのさまざまな数学的モデルが使われている。データマイニングは次のように定義することができる: “(多くの場合大規模) データの分析であり、斬新な方法で思いも寄らない関係を発見しデータを要約するもので、その結果をデータの所有者が理解でき有益である”。多種多様な技術の開発されている: 分類 (例えば、決定木学習)、回帰、クラスタリング (例えば、k-means クラスタリング)、パターン発見 (例えば、相関ルール学習)。

両方のフィールド (運用管理とデータマイニング) からは貴重な分析技術が得られる。ここでの課題は、プロセスマイニングとこれらのフィールドの技術を組み合わせることである。シミュレーションを考えてみると、プロセスマイニング技術は、履歴データに基づいてシミュレーションモデルを学習するのに使える。また、シミュレーションモデルは運用サポートに用いることができる。イベントログとモデルは密接に関係するので、モデルは過去を再生するために使用されたり、現状データに基づいて現在の状態からシミュレーションを行い未来への “早送りボタン” を提供することができる。

同じように、プロセスマイニングと**視覚的な分析**を一体化することが望ましい。視覚的な分析は自動分析とインタラクティブな可視化を組み合わせたもので、大規模かつ複雑なデータセットをより良く理解するのに役立つ。視覚的な分析は非構造化データ内のパターンを理解する人間の驚くべ

き能力を有効に利用している。自動プロセスマイニング手法をインタラクティブな可視化分析と組み合わせることで、イベントデータからより多くの洞察を抽出することが可能となる。

4.10 C10: 非専門家向けにユーザビリティを向上する

プロセスマイニングの目標の一つは、“生きているプロセスモデル”、すなわち、記録されて終わる静的モデルではなく、日常的に使用されるプロセスモデル、を作成することである。新しいイベントデータは、新たに発生した挙動を発見するのに使える。イベントデータとプロセスモデルの間には関連があるので、現在の状態と最近の活動を最新のモデル上に表示できる。したがってエンドユーザは日常的にプロセスマイニングの結果を操作できる。このような操作性は非常に重要であるが、直感的なユーザーインターフェースも求められる。ここでの課題は、洗練されたプロセスマイニングアルゴリズムが自動的にパラメータを設定し、適切なタイプの分析を提案することができ、それをユーザーフレンドリーなインターフェイスの背後に配置して利用者に意識させないようにすることである。

4.11 C11: 非専門家向けに理解度を向上する

プロセスマイニングで結果を生成することが容易でも、結果が実際に有用であるとは限らない。利用者が出力結果を理解しにくかったり、誤った結論に至ってしまうかもしれない。このような問題を回避するためには、結果が適切な表現で提示されるべきである (GP5 も参照)。また、結果の信頼性がいつも明確に示されるべきである。場合によっては特定の結論を正当化するにはデータが少なすぎることがある。実際、既存のプロセス発見技術ではアンダーフィッティングやオーバーフィッティングの警告が表示されず、データが少なすぎ結論を正当化できないことが明らかでも、モデルを表示してしまう。

5 エピローグ

IEEE プロセスマイニングタスクフォースは、(a) プロセスマイニングの適用を促進する、(b) ソフトウェア開発者、コンサルタント、経営者、エンドユーザが最先端の技術を使うのを支援する、(c) プロセスマイニングの研究を促進する、ことを目指している。本マニフェストでは、タスクフォースの原則と意図を述べた。プロセスマイニングのトピックを紹介し、指針 (3 節) と課題 (4 節) の一覧を示した。指針を用いて、明らかな間違いを避けることができる。課題リストでは研究開発の方向性を示した。両者ともプロセスマイニングの成熟度レベルを上げることを目指している。

最後に、用語を補足する。プロセスマイニングの分野では以下の用語が使われる: ワークフローマイニング、(ビジネス) プロセスマイニング、自動 (ビジネス) プロセス発見、(ビジネス) プロセスインテリジェンス。異なる組織が重複する概念に対して異なる用語を使っている。例えば、Gartner は“自動ビジネスプロセス発見 (Automated Business Process Discovery)” (ABPD) を、Software AG は“プロセスインテリジェンス” をそれぞれ用語として広めている。用語 “ワークフローマイニング” は、ワークフローモデルの作成がプロセスマイニングの適用のひとつにすぎないので、あまり適切ではない。同様に、用語 “ビジネス” を追加すると、プロセスマイニングの適用範囲を狭くすることになる。つまり “ビジネス” を追加すると不適切になってしまう適用が多くある (例えば、先端技術システムの利用分析や、ウェブサイト分析)。プロセス発見はプロセスマイニング領域の重要な部分であるが、多くあるユースケースのひとつに過ぎない。適合性検査、予測、組織に関するマイニング、ソーシャルネットワーク分析、等は、プロセス発見の枠を超える、別のユースケースである。

図 7 は前述の用語のいくつかを関連付けたものである。意思決定の支援に使える実用的な情報を提供することを目指すすべての技術や手法は、ビジネスインテリジェンス (BI) の下に配置することができる。(ビジネス) プロセスインテリジェンスは、BI と BPM を結合したものと見ることができる。つまり、BI 技術はプロセスとプロセス管理を分析・改善するために用いられる。プロセスマイニングは、イベントログを出発点とするプロセスインテリジェンスの具体化と考えられる。(自動ビジネス) プロセス発見は、プロセスマイニングの三つの基本タイプの一つにすぎない。ほとんどの BI ツールが本マニフェストで述べたようなプロセスマイニング機能を提供していないという意味では、図 7 は少し誤解を招く可能性がある。また、広範な BI 領域のほんの少しの部分をカバーしているようなツールや手法に対しても、用語 “BI” が使われている。

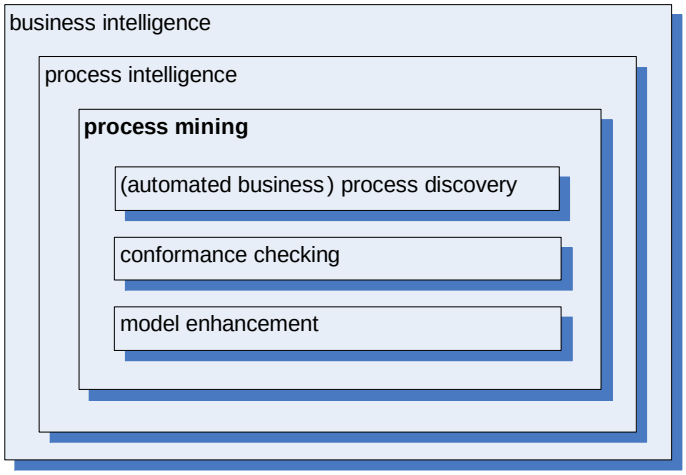

図 7. 異なる用語を関連づける。

別の用語を使うのは、商業上の理由かもしれない。ベンダーは特定の特徴 (例えば、発見やインテリジェンス) を強調したいときがある。しかし混乱を避けるために、本マニフェストがカバーしている分野に対しては、用語 “プロセスマイニング” を使用するのが望ましい。

用語集

- **Activity (アクティビティ)**: プロセスにおける明確に定義されたステップ。イベントは、特定のプロセスインスタンスにおけるアクティビティの開始、完了、キャンセル、等に対応する。
- **Automated Business Process Discovery (自動ビジネスプロセス発見)**: → **Process Discovery (プロセス発見)**
- **Business Intelligence (ビジネスインテリジェンス)** (BI): 意思決定を支援するためにデータを使用する、ツールと方法の広範な集合。
- **Business Process Intelligence (ビジネスプロセスインテリジェンス)**: → **Process Intelligence (プロセスインテリジェンス)**
- **Business Process Management (ビジネスプロセスマネジメント)** (BPM): 情報技術の知識と経営科学の知識を融合し、両方を業務プロセスに適用する分野。
- **Case (事例)**: → **Process Instance (プロセスインスタンス)**
- **Concept Drift (コンセプトドリフト)**: プロセスが時間の経過とともに変化する現象。プロセスが季節の変化や競争の激化によって徐々に (または突然) 変化するもので、分析が複雑になる。
- **Conformance Checking (適合性検査)**: ログに記録されている現実がモデルと適合するか、また逆もそうかを分析する。目標は不一致を検出しその重大性を評価することである。適合性検査は、プロセスマイニングの三つの基本タイプの一つである。
- **Cross-Organizational Process Mining (組織横断的なプロセスマイニング)**: 複数の組織が起源であるイベントログへのプロセスマイニング技術の適用。
- **Data Mining (データマイニング)**: (多くの場合、大規模な) データ集合の分析であり、新たな洞察を提供するようなやり方で、予想外の関係を見つけデータを要約する。
- **Event (イベント)**: ログに記録された行動で、例えば、あるプロセスインスタンスのアクティビティの開始、完了、キャンセル。
- **Event Log (イベントログ)**: プロセスマイニング入力として使われるイベントの集合。イベントは、個別のログファイルに格納する必要はない。(例えば、イベントが別々のデータベースのテーブルに散在している場合がある)。
- **Fitness (フィットネス)**: 与えられたモデルがイベントログに見られる挙動をどのくらい良く許容するかを判断する評価基準。もしログ内のすべてのトレースが、最初から最後まで、モデルによって再生できる場合、モデルは完璧なフィットネスを持っている。

- **Generalization (汎化)**: モデルが観測されていない挙動をどのくらい良く許容できるかを判断する評価基準。“オーバーフィッティング” モデルは、十分に汎化することができない。
- **Model Enhancement (モデル強化)**: プロセスマイニングの三つの基本タイプの一つ。ログから抽出した情報を用いて、プロセスモデルが拡張または改善される。例えば、タイムスタンプを調べながらプロセスモデル上でイベントログを再生するとボトルネックを見つけることができる。
- **MXML**: イベントログをやり取りするための XML ベースのフォーマット。MXML の代わりとして、新しいツール非依存のプロセスマイニング用フォーマットである XES がある。
- **Operational Support (運用サポート)**: 実行中のプロセスインスタンスを監視し影響を与えることを目指した、イベントデータのオンライン分析。3 種類の運用サポート活動がある: **検出** (観測された挙動がモデル化された挙動から逸脱した場合に、アラートを生成する)、**予測** (過去の挙動に基づいて将来の挙動を予測する、例えば残りの処理時間を予測)、**推奨** (特定の目標 (例えばコストを最小にする) を実現するために適切な行動を提案する)。
- **Precision (精度)**: イベントログに見られる挙動とは大きく異なる挙動をモデルが禁止するかどうかを判断する評価基準。低精度のモデルは “アンダーフィッティング” である。
- **Process Discovery (プロセス発見)**: プロセスマイニングの三つの基本タイプの一つ。イベントログに基づいて、プロセスモデルが学習される。例えば、α アルゴリズムは、イベントの集合に含まれているプロセスのパターンを識別し、ペトリネットを発見することができる。
- **Process Instance (プロセスインスタンス)**: 分析対象のプロセスによって処理されるエンティティ。イベントはプロセスインスタンスに属する。プロセスインスタンスの例は、顧客注文、保険金請求、融資申し込み、などである。
- **Process Intelligence (プロセスインテリジェンス)**: ビジネスプロセスマネジメントに焦点を当てたビジネスインテリジェンスの一種。
- **Process Mining (プロセスマイニング)**: 実際のプロセス (想定のプロセスではない) を発見、監視、改善するための技術、ツール、方法であり、今日の (情報) システムで普通に入手可能なイベントログから抽出した知識を用いる。
- **Representational Bias (表現バイアス)**: プロセスマイニングの結果を提示し構築するために選択された、ターゲット言語。
- **Simplicity (単純さ)**: オッカムの剃刀が適用できるようにする評価基準。すなわち、ログに見られる挙動を説明できる最も単純なモデルは、最高のモデルである。単純さは、さまざまな方法で定量化できる (例えば、モデル内のノードとアークの数)。
- **XES**: イベントログのための XML ベースの規格である。本標準は、イベントログのデフォルトの交換形式として IEEE プロセスマイニングタスクフォースで採用されている (www.xes-standard.org を参照のこと)。