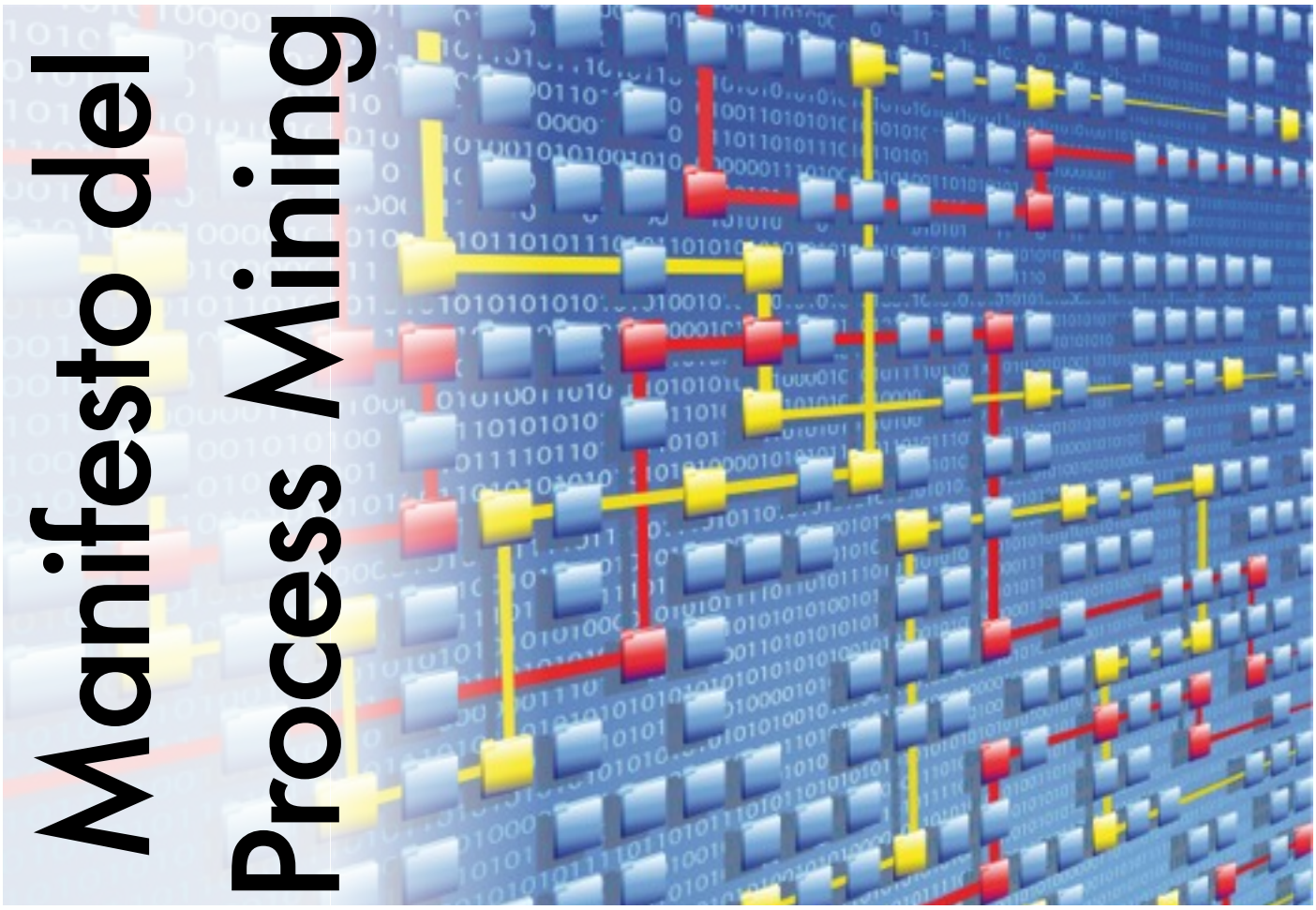
Manifesto del Process Mining



Un *manifesto* è una "dichiarazione pubblica di principi ed intenti" redatta da un gruppo di persone. Questo manifesto è stato scritto dai membri e dai sostenitori della *IEEE Task Force* sul *Process Mining*. L'obiettivo della *task force* è quello di promuovere la ricerca, lo sviluppo, la divulgazione, l'implementazione, l'evoluzione e la comprensione del *Process Mining*.

Il Process Mining costituisce un'area di ricerca relativamente giovane, che si trova, da un lato, tra la computational intelligence ed il data mining e, dall'altro, tra la modellazione e l'analisi dei processi. L'idea di base del Process Mining è quella di dedurre, monitorare e migliorare i processi reali (cioè non ipotetici) estraendo conoscenza dai log, oggi ampiamente disponibili nei sistemi informativi (si veda la Fig. 1). Il Process Mining include la deduzione (automatica) di processi (discovery), cioè l'estrazione di un modello di processo a partire da un log; la verifica di conformità (conformance checking), cioè il monitoraggio di eventuali discrepanze tra un modello e un log; l'individuazione di reti sociali (social network) e organizzative; la costruzione automatica di modelli di simulazione; l'estensione e la revisione di modelli; la predizione delle possibili future evoluzioni di un'istanza di processo; le raccomandazioni su come operare sulla base di dati storici. Il Process Mining fornisce un importante

Contenuto:

Le tecniche di *Process Mining* consentono di *estrarre conoscenza dai log* che vengono registrati dai sistemi informativi odierni. Grazie a queste tecniche è possibile *dedurre, monitorare, e migliorare i processi* in una grande varietà di domini applicativi. Il crescente interesse per il *Process Mining* è dovuto essenzialmente a due motivi: da un lato, all'enorme disponibilità di dati che forniscono informazioni dettagliate sulle passate esecuzioni dei processi; dall'altro, alla necessità di migliorare e supportare i processi aziendali in ambienti competitivi e in rapida evoluzione. Questo manifesto è stato creato dalla *IEEE Task Force sul Process Mining* con lo scopo di promuovere l'interesse per questa disciplina. Inoltre, definendo un insieme di principi guida e individuando nuove importanti sfide in questo campo, il manifesto rappresenta un'ottima *guida per sviluppatori di software, ricercatori, consulenti, business managers*, e per svariati altri tipi di utenti. Obiettivo ultimo di questo manifesto è quello di incrementare il livello di maturità del *Process Mining* quale nuovo strumento per migliorare, (ri)modellare, monitorare, e supportare i processi operativi aziendali.

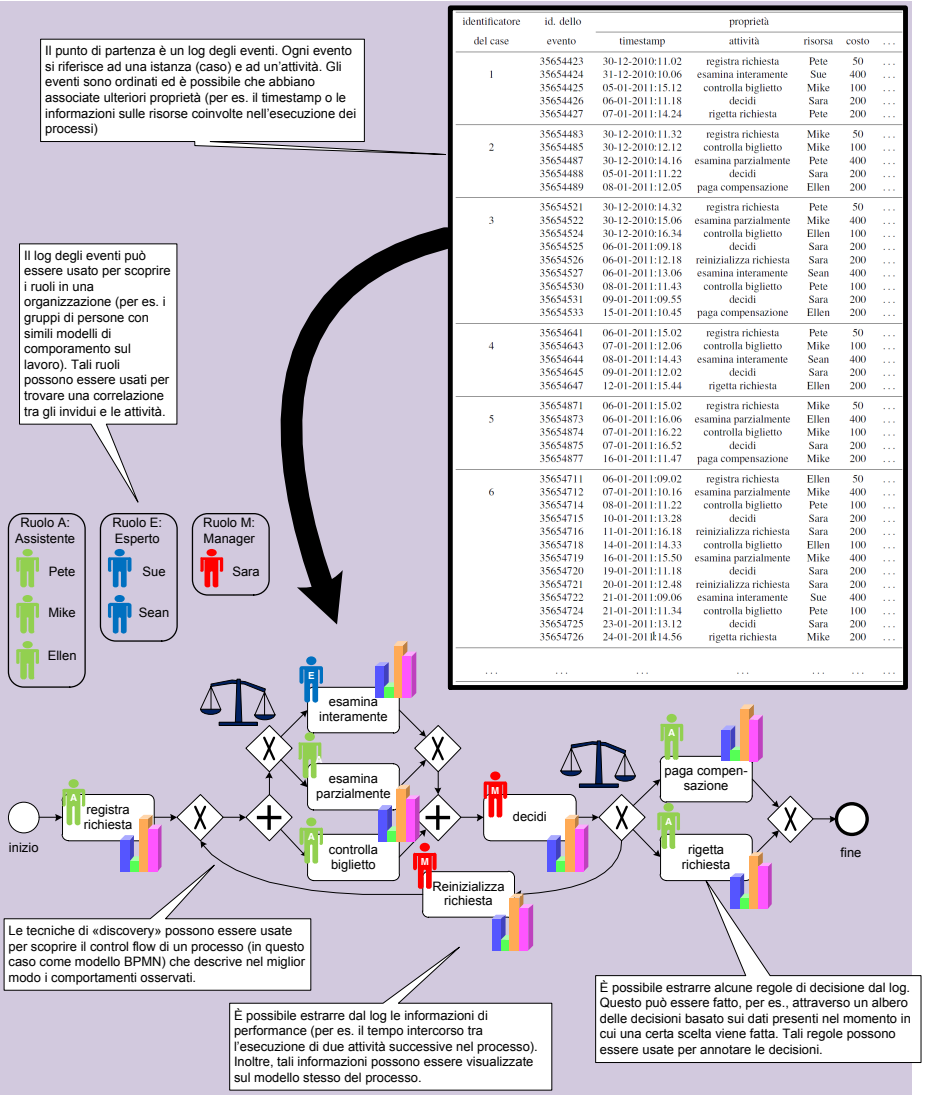


Figura 1: Le tecniche di Process Mining estraggono conoscenza a partire da log di eventi al fine di scoprire, monitorare e migliorare i processi.

Alcuni obiettivi concreti della task force sono:

1) rendere noto lo stato dell'arte sul *Process Mining* a utenti, sviluppatori, consulenti, *business manager* e ricercatori;
2) promuovere l'uso delle tecniche e degli strumenti di *Process Mining* e stimolarne nuove applicazioni;
3) partecipare attivamente alla standardizzazione di come gli eventi vengono memorizzati nei *log*;
4) organizzare tutorial, special session, workshop, panels, e
5) pubblicare articoli, libri, video ed edizioni speciali di riviste.

punto di contatto fra data mining e modellazione e analisi di processi. Nella Business Intelligence (BI) sono stati introdotti molti termini che si riferiscono a sistemi di reportistica e "console elettroniche" piuttosto semplici. Il termine Business Activity Monitoring (BAM), per esempio, si riferisce a tecnologie che consentono il monitoraggio in tempo reale di processi aziendali. Il Complex Event Processing (CEP) si riferisce invece a tecnologie per l'analisi di grandi quantità di eventi e l'uso di questi per il monitoraggio, l'indirizzamento e l'ottimizzazione in tempo reale del business. Corporate Performance Management (CPM) è ancora un altro neologismo che si riferisce alla misurazione delle performance di un processo o di un'organizzazione. Altri approcci legati al management dei processi sono il Continuous Process Improvement (CPI), il Business Process Improvement (BPI), il Total Quality management (TQM), e il Six Sigma. Tutti questi approcci condividono l'idea che un processo vada "analizzato al microscopio" per identificare possibili miglioramenti. Il Process Mining rende di fatto possibili tutti questi approcci.

Mentre gli strumenti di BI e gli approcci per il management, quali Six Sigma e TQM, mirano a migliorare le performance "operative", come per esempio ridurre il tempo di esecuzione o eliminare altri tipi di difetti in un processo, le aziende stanno iniziando a porre molta enfasi sulla corporate governance, sui rischi, e sulla conformità. Regolamentazioni, quali il Sarbanes-Oxley Act (SOX) e l'accordo Basilea II, per esempio, trattano problemi riguardanti la conformità. Le tecniche di Process Mining offrono strumenti per rendere i controlli di conformità più rigorosi e per accertare la validità e l'affidabilità delle informazioni riguardanti i processi chiave di un'azienda.

Nel corso degli ultimi decenni è aumentata la disponibilità e la reperibilità di grosse quantità di dati e le tecniche di Process Mining hanno avuto l'opportunità di maturare. Inoltre, come appena detto, approcci al management legati al miglioramento dei processi (come per esempio Six Sigma, TQM, CPI, e CPM) e legati alla conformità (come per esempio SOX, BAM, ecc.) possono beneficiare dello sviluppo del Process Mining. Fortunatamente, gli algoritmi di Process Mining sono stati implementati in diversi sistemi, sia accademici che commerciali. Ad oggi, esiste un attivo gruppo di ricercatori che lavora sullo sviluppo di tecniche di Process Mining che pertanto sta diventano uno degli "argomenti caldi" nella ricerca sul Business Process Management (BPM). Oltre a ciò, si sta sviluppando un forte interesse da parte delle aziende nei riguardi di temi legati al Process Mining e sempre più i produttori di software stanno introducendo funzionalità di Process Mining nei loro prodotti. Esempi di tali prodotti sono: ARIS Process Performance Manager (Software AG), Comprehend (Open Connect), Discovery Analyst (StereoLOGIC), Flow (Fourspark), Futura Reflect (Futura Process

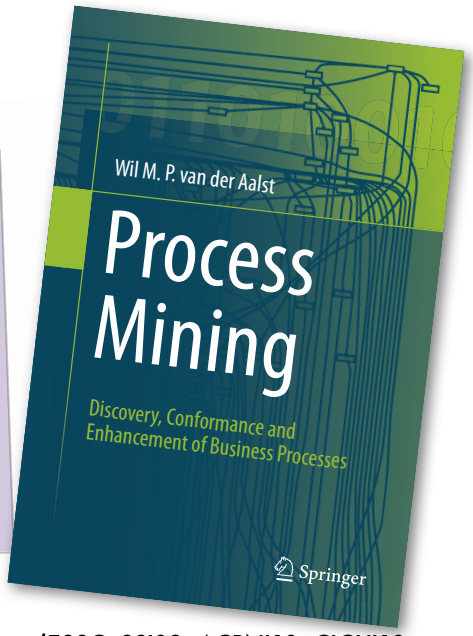
Intelligence), Interstage Automated Process Discovery (Fujitsu), OKT Process Mining suite (Exeura), Process Discovery Focus (Iontas/Verint), ProcessAnalyzer (QPR), ProM (TU/e), Rbminer/Dbminer (UPC), e Reflect|one (Pallas Athena).

Il crescente interesse nell'analisi dei processi basata sui log ha motivato la formazione della Task Force sul Process Mining. La task force è stata creata nel 2009, nel contesto del Data Mining Technical Committee (DMTC) della Computational Intelligence Society (CIS) e dell'Institute of Electrical and Electronic Engineers (IEEE). Attualmente, la task force è composta da membri che rappresentano rivenditori software (e.g. Pallas Athena, Software AG, Futura Process Intelligence, HP, IBM, Infosys, Fluxicon, Businesscape, Iontas/Verint, Fujitsu, Fujitsu Laboratories, Business Process Mining, Stereologic), aziende di consulenza/utenti (e.g. ProcessGold, Business Process Trends, Gartner, Deloitte, Process Sphere, Siav SpA, BPM Chili, BWI Systeme GmbH, Excellentia BPM, Rabobank), e istituti di ricerca (e.g. TU/e, Università degli Studi di Padova, Universitat Politècnica de Catalunya, New Mexico State University, IST - Technical University of Lisbon, Università della 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, Università di Bologna, Ulsan National Institute of Science and Technology, Cranfield University, K.U. Leuven, Tsinghua University, University of Innsbruck, University of Tartu).

Fin dalla sua fondazione nel 2009, la task force ha svolto varie attività riconducibili agli obiettivi sopradescritti, tra i quali, vari workshop e special session come il workshop sulla Business Process Intelligence (BPI'09, BPI'10, e BPI'11) e special track a importanti conferenze IEEE (come per esempio CIDM'11). Il bagaglio di conoscenze sul Process Mining è stato diffuso tramite tutorial (come per esempio WCCI'10 e PMPM'09), scuole estive (ESSCaSS'09, ACPN'10, CICH'10, etc.), video (cfr. www.processmining.org), e numerose pubblicazioni, tra cui il primo libro sul Process Mining, recentemente pubblicato dalla casa editrice Springer. La task force ha anche co-organizzato la prima Business Process Intelligence Challenge (BPIC'11), una competizione dove i partecipanti dovevano estrarre conoscenza da un log di grandi dimensioni e complesso. Nel 2010, la task force ha anche standardizzato XES (www.xes-standard.org), un formato per la memorizzazione dei log, estensibile e supportato dalla libreria OpenXES (www.openxes.org) e da strumenti quali ProM, XESame, Nitro, etc. All'indirizzo www.win.tue.nl/ieeetfpm/ è possibile reperire maggiori informazioni circa le attività della task force.

Il Libro sul Process Mining

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



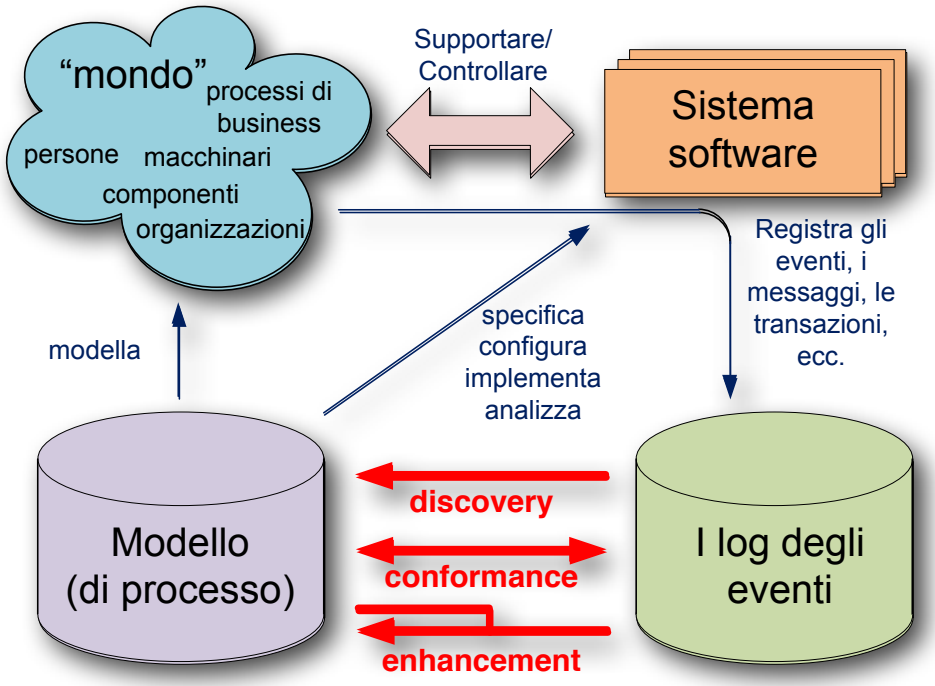


Figura 2: I tre tipi principali di *Process Mining*: (a) *discovery*, (b) *conformance checking*, e (c) *enhancement*.

2. Process Mining: Stato Dell'arte

La capacità di espansione dei sistemi informativi e di altri sistemi computazionali sono caratterizzate dalla legge di Moore. Gordon Moore, il cofondatore di Intel, aveva previsto, nel 1965, che il numero di componenti in circuiti integrati sarebbe raddoppiato ogni anno. Durante gli ultimi 50 anni questa crescita è stata effettivamente esponenziale sebbene leggermente più lenta. Questi progressi hanno condotto ad un'incredibile espansione dell'"universo digitale" (cioè tutti i dati immagazzinati e/o scambiati elettronicamente). Inoltre, col passare del tempo, universo digitale e universo reale si stanno via via allineando.

La crescita di un universo digitale

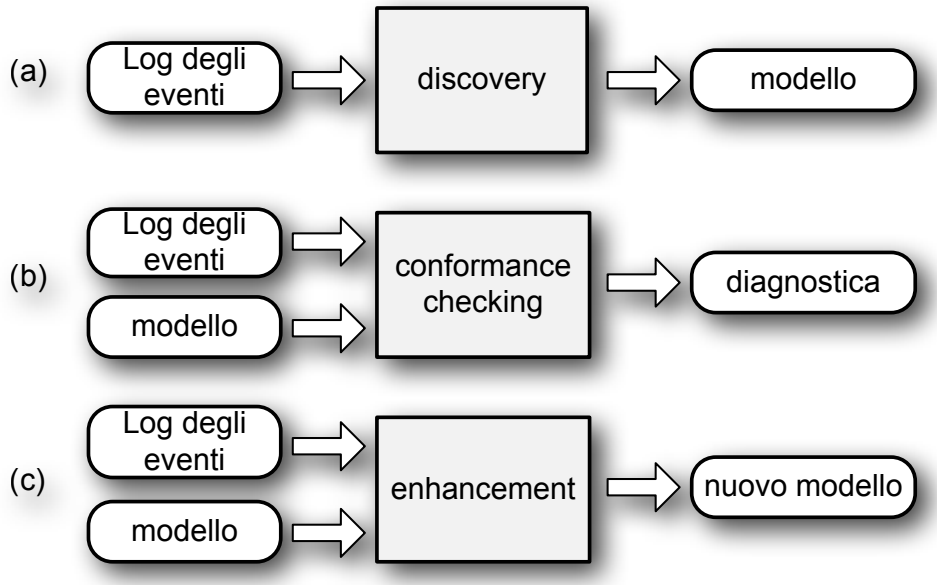


Figura 3: I tre tipi base di *Process Mining* spiegati in termini di *input* e *output*: (a) *discovery*, (b) *conformance checking*, ed (c) *enhancement*.

che risulti ben allineato con i processi nelle organizzazioni rende possibile registrare ed analizzare *eventi*. Gli eventi possono essere di vario tipo: da un utente che ritira del denaro contante da uno sportello automatico, a un dottore che regola un apparecchio per i raggi X, a un comune cittadino che fa richiesta per una patente di guida, a un contribuente che sottomette una dichiarazione dei redditi, a un viaggiatore che riceve il numero di un biglietto elettronico. La nuova sfida è cercare di sfruttare i dati in modo significativo, per esempio per fornire suggerimenti, identificare colli di bottiglia, prevedere problemi, registrare violazioni di regole, raccomandare contromisure e dare "forma" ai processi. Lo scopo del *Process Mining* è fare esattamente questo.

Il punto di partenza per qualsiasi tecnica di *Process Mining* è un *log* degli eventi (*event log* o semplicemente *log*). Tutte le tecniche di *Process Mining* assumono che sia possibile registrare *sequenzialmente eventi* in modo che ogni evento si riferisca ad una determinata *attività* (cioè ad un passo ben definito di un processo) e sia associato ad un particolare *case* (cioè un'istanza di processo). I *log* possono contenere anche ulteriori informazioni circa gli eventi. Di fatto, qualora sia possibile, le tecniche di *Process Mining* usano informazioni supplementari come le *risorse* (cioè le persone o i dispositivi) che eseguono o che danno inizio ad un'attività, i *timestamp* o altri dati associati ad un evento (per esempio la dimensione di un ordine).

Come mostrato in Fig. 2, gli *event log* possono essere usati per eseguire tre tipi di *Process Mining*, il primo dei quali è detto *discovery*. Una tecnica di *discovery* prende in *input* un *event log* e produce un modello senza utilizzare alcuna informazione a priori. Il *process discovery* è la più importante tecnica di *Process Mining*, e molte organizzazioni che ne hanno avuto esperienza trovano stupefacente come queste tecniche possano effettivamente descrivere processi reali solamente basandosi su esempi di esecuzione. Il secondo tipo di *Process Mining* è il *conformance checking*. In questo caso, un modello di processo preesistente è confrontato con informazioni (relative allo stesso processo) estratte da un *event log*. Il *conformance checking* può essere usato per verificare se ciò che accade nella realtà (come risulta dai *log*) è conforme al modello e viceversa. Il *conformance checking* può essere applicato a diversi tipi di modelli: procedurali, organizzativi, dichiarativi, regole di *business*, leggi, etc. Il terzo tipo di *Process Mining* è l'*enhancement*. In tal caso, l'idea è quella di estendere o migliorare un modello di processo esistente usando informazioni circa il processo contenute nei *log*. Mentre il *conformance checking* misura quanto un modello è allineato con ciò che accade nella realtà, questo terzo tipo di *Process Mining* si propone di cambiare o estendere il modello preesistente. Per esempio usando i

Elementi caratterizzanti il Process Mining:

1. Il Process Mining non si limita al discovery del flusso di controllo.
Il *discovery* di modelli di processo da un *log* piace molto a professionisti e accademici. Per questo motivo, il *discovery* del flusso di controllo è spesso visto come la parte più interessante del *Process Mining*. Tuttavia, il *Process Mining* non si limita al discovery del flusso di controllo. Da un lato, il *discovery* è solo una delle tre forme possibili del *Process Mining* (assieme a *conformance* e *enhancement*) e, dall'altro, il focus del *discovery* non si limita al flusso di controllo: la prospettiva organizzativa, di istanza e temporale svolgono un ruolo altrettanto importante.

2. Il Process Mining non è solo un particolare tipo di data mining.
Il Process Mining può essere considerato come "l'anello mancante" fra il *data mining* ed il tradizionale *model-driven* BPM. La maggior parte delle tecniche di *data mining* non sono infatti orientate ai processi. I modelli di processo possono esprimere comportamenti concorrenti che sono incomparabili con le strutture che tipicamente caratterizzano il *data mining*, quali alberi di decisione e regole associative. Per questi motivi, è necessario stabilire modelli di rappresentazione e algoritmi completamente nuovi.

3. Il Process Mining non si limita ad una analisi offline.
Le tecniche di *Process Mining* estraggono conoscenza a partire da dati storici. Anche se è possibile effettuare analisi a partire da dati "post mortem", il risultato di queste può essere applicato per avere informazioni circa le istanze in esecuzione. Ad esempio, è possibile predire il tempo di completamento di un ordine iniziato da un cliente, sulla base di un modello di processo precedentemente estratto da un *log*.

timestamp in un *event log* è possibile estendere il modello per mostrare colli di bottiglia, livelli di un servizio, tempi di produttività e frequenze.

La Fig. 3 descrive i tre tipi di *Process Mining* in termini di *input/output*. Le tecniche di *discovery* prendono in *input* un *event log* e producono un modello. Il modello estratto è tipicamente un modello di processo (per esempio una rete di Petri, un modello BPMN, un modello EPC o un diagramma UML delle attività). Tuttavia, il modello può anche descrivere altre prospettive (come per esempio una *social network*). Le tecniche di *conformance checking* prendono in *input* un *event log* e un modello. L'*output* consiste in una serie di informazioni diagnostiche che mostrano le differenze tra il modello e il *log*. Anche le tecniche di *enhancement*, infine, (revisione o estensione) richiedono un *event log* e un modello in *input*. L'*output* è il modello stesso migliorato o esteso.

Il *Process Mining* include varie prospettive. La *prospettiva del flusso di controllo* si focalizza sull'ordine delle attività. L'obiettivo di questa prospettiva consiste nel trovare una buona caratterizzazione di tutti i possibili percorsi in un processo. Il risultato, tipicamente, è espresso in termini di una rete di Petri o di altri formalismi (come per esempio EPC, BPMN o diagrammi UML delle attività). La *prospettiva dell'organizzazione* si focalizza sulle informazioni che riguardano le risorse contenute (e spesso non visibili) all'interno dei *log*, ovvero, quali attori (per esempio persone, sistemi, ruoli o dipartimenti) sono coinvolti e come questi si relazionano fra loro. L'obiettivo è quello di strutturare l'organizzazione classificando le persone in base ai ruoli che svolgono e alle unità organizzative, oppure di costruire una rappresentazione della *social network* dell'organizzazione. La *prospettiva dell'istanza* si concentra sulle proprietà di un *case*. Ovviamente, un *case* può essere caratterizzato dal suo percorso nel processo oppure dagli attori che operano nello stesso. Tuttavia, i *case* possono essere definiti anche attraverso valori assunti da altri tipi di dati. Ad esempio, se un *case* rappresenta la compilazione di un ordine, potrebbe essere interessante conoscere il fornitore o il numero di prodotti ordinati. La *prospettiva del tempo* è legata a quando un evento è accaduto e alla sua frequenza. Quando agli eventi sono associati dei *timestamp*, è possibile individuare colli di bottiglia, misurare i livelli di un servizio, monitorare l'uso delle risorse e predire il tempo restante per il completamento di un'istanza.

Esistono alcuni fraintendimenti comuni quando si parla di Process Mining. Alcuni produttori, analisti e ricercatori tendono a pensare alle tecniche di Process Mining come a a speciali approcci di data mining per il discovery di processi che si limitano ad un'analisi offline. Tuttavia, questa visione non è corretta. Pertanto, evidenziamo tre elementi caratterizzanti il *Process Mining* che sono sintetizzati nel riquadro della pagina precedente.

Per contestualizzare il *Process Mining*, consideriamo il ciclo di vita del *Business Process Management* (BPM), presentato in Fig. 4. Il ciclo di vita BPM mostra le sette fasi di un processo di *business* ed i sistemi informativi sottostanti. Nella *fase di (ri)modellazione* si crea un nuovo modello di processo, oppure se ne aggiorna uno preesistente. Nella *fase di analisi*, un processo candidato ed una possibile alternativa sono messi a confronto. Terminata la fase di (ri)modellazione, si implementa il modello (*fase di implementazione*) oppure, si (ri)configura un sistema informativo già esistente (*fase di (ri)configurazione*). Nella *fase di esecuzione*, il modello di processo è eseguito. Durante l'esecuzione, il processo è monitorato. Minori adattamenti sono possibili (*fase di adattamento*) senza che sia realizzata una nuova modellazione del processo. Nella *fase di diagnosi*, l'esecuzione del processo viene analizzata e questo può portare ad una nuova fase di rimodellazione. Il *Process Mining* è uno strumento utile per la maggior parte della fasi descritte in Fig. 4. Ovviamente, la fase di diagnosi può beneficiare maggiormente delle tecniche di *Process Mining* che però tuttavia non si limitano al supporto di questa sola fase. Ad esempio, durante l'esecuzione, le tecniche di *Process Mining* possono essere utilizzate come

Principi guida:

PG1: gli eventi devono essere trattati come entità di prima classe

PG2: L'estrazione dei log deve essere guidata da domande

PG3: Occorre supportare concorrenza, punti di scelta e altri costrutti di base legati al flusso di controllo

PG4: gli eventi devono essere legati ad elementi del modello

PG5: i modelli devono essere trattati come astrazioni utili della realtà

PG6: il Process Mining deve essere un processo continuo

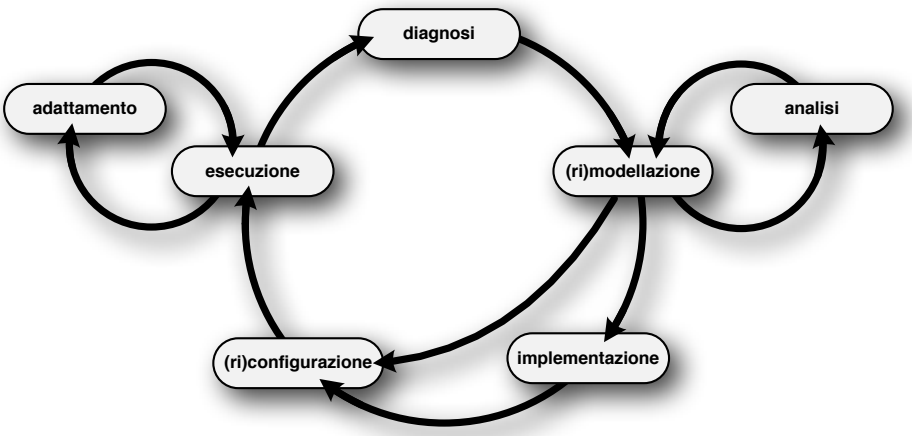


Figura 4: Il ciclo di vita BPM, che identifica le varie fasi di un processo ed i sistemi informativi sottostanti; il Process Mining può essere coinvolto in tutte le fasi (ad eccezione dell'implementazione).

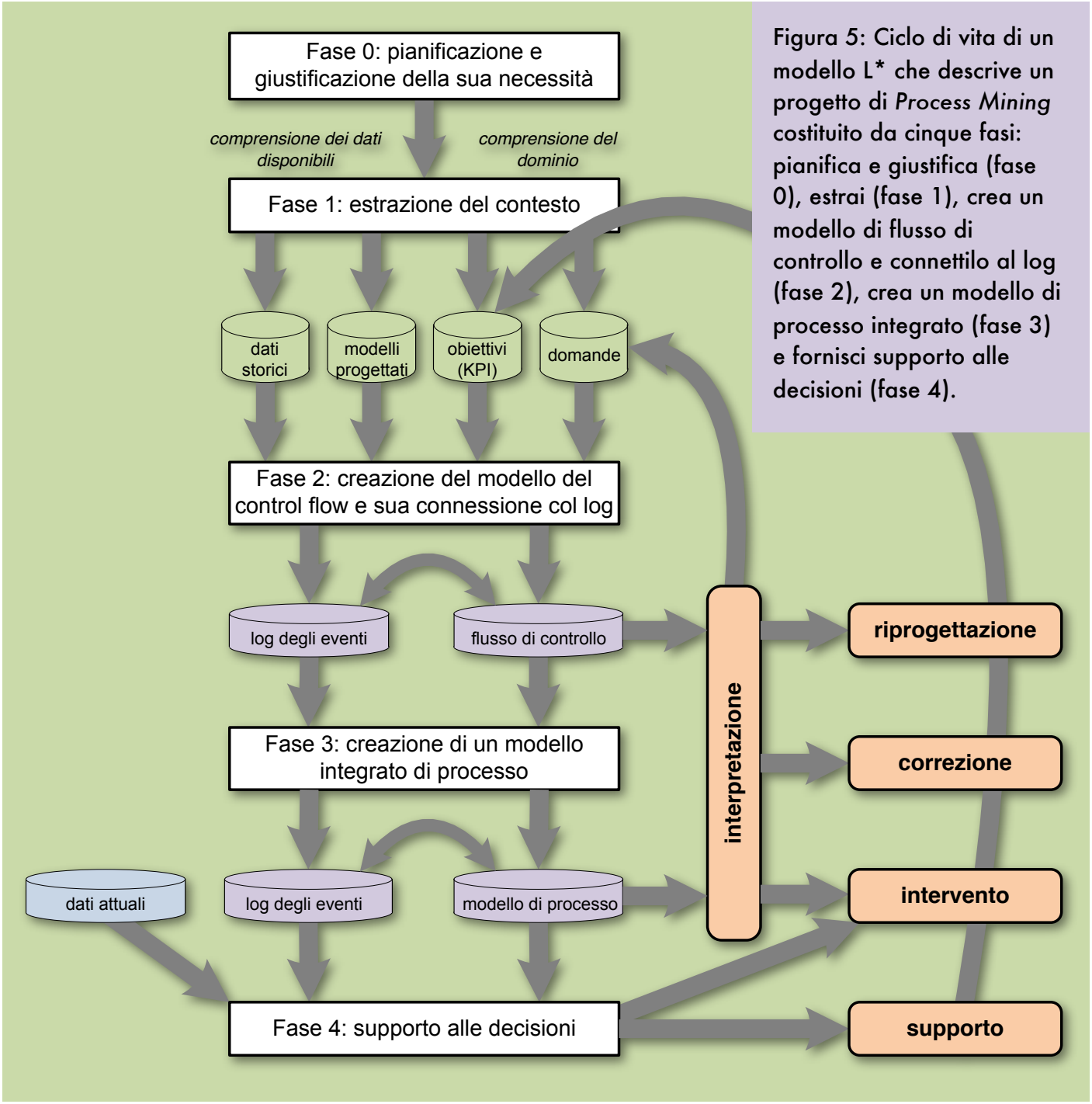


Figura 5: Ciclo di vita di un modello L* che descrive un progetto di *Process Mining* costituito da cinque fasi: pianifica e giustifica (fase 0), estrai (fase 1), crea un modello di flusso di controllo e connettilo al log (fase 2), crea un modello di processo integrato (fase 3) e fornisci supporto alle decisioni (fase 4).

supporto alle decisioni (*operational support*). Si possono utilizzare per esempio strumenti di predizione e raccomandazione (che apprendono modelli sulla base di dati storici) per modificare le istanze di processo in esecuzione. Forme simili di supporto alle decisioni possono aiutare ad adattare il processo e a guidare la rimodellazione dello stesso.

Mentre la Fig. 4 presenta il ciclo di vita di BPM, la Fig. 5 si focalizza sulle effettive attività e sui risultati del *Process Mining*. In Fig. 5 sono riportate le possibili fasi in cui si può trovare un progetto di *Process Mining*. Ogni progetto inizia con una pianificazione ed una giustificazione dell'attività di pianificazione stessa (fase 0). Dopo l'avvio del progetto, è necessario interrogare sistemi informativi, esperti di dominio e *manager* per ricavare dati, modelli, obiettivi e domande a cui è necessario successivamente rispondere (fase 1). Questa attività richiede una comprensione dei dati che si hanno a disposizione ("quali dati si possono usare per l'analisi?") e del dominio ("quali sono le domande rilevanti?") e fornisce i risultati riportati in Fig. 5 (cioè dati storici, modelli progettati, obiettivi, domande). Durante la fase 2, si costruisce il modello del flusso di controllo e lo si collega al *log*. In questa fase si possono usare tecniche di *discovery* automatico. Il modello dedotto può già fornire risposta ad alcune delle domande poste e potrebbe portare ad adattamenti e rimodellazioni. Inoltre, il *log* può essere filtrato o adattato sulla base del modello (ad esempio, rimuovendo attività rare o istanze anomale ed inserendo eventi mancanti). Talvolta ulteriore lavoro è richiesto per correlare gli eventi appartenenti alla stessa istanza di processo. Gli eventi rimanenti sono correlati ad entità del modello di processo. Quando il processo è piuttosto strutturato, il modello del flusso di controllo può essere esteso con altre prospettive (ad esempio, dati, tempi, risorse) durante la fase 3. La

Livello	Descrizione	Esempi
★★★★★	Livello più elevato: l'*event log* è di qualità eccellente (ovvero affidabile e completo) e gli eventi sono ben definiti. Gli eventi vengono registrati con un approccio automatico, sistematico, attendibile e sicuro. Considerazioni legate a *privacy* e sicurezza sono affrontate adeguatamente. Inoltre, gli eventi registrati (e tutti i loro attributi) hanno una semantica chiara. Questo implica l'esistenza di una o più ontologie, a cui gli eventi e gli attributi si riferiscono.	*Log* di sistemi BPM annotati semanticamente.
★★★★	Gli eventi vengono registrati automaticamente e in modo sistematico ed attendibile, ovvero il *log* è affidabile e completo. A differenza dei sistemi che operano al livello ★★★, concetti quali istanza di processo (*case*) e attività sono supportati in maniera esplicita.	Gli *event log* dei classici sistemi di BPM e *workflow*.
★★★	Gli eventi vengono tracciati automaticamente, ma non viene seguito nessun approccio sistematico. A differenza dei *log* a livello ★★, c'è un certo grado di garanzia che gli eventi registrati corrispondano alla realtà (in altre parole, l'*event log* è affidabile ma non necessariamente completo). Si considerino, a scopo illustrativo, gli eventi generati da un sistema ERP. Anche se questi eventi devono essere estratti da diverse tabelle, si può assumere che l'informazione ottenuta sia corretta (per esempio è sicuro che un pagamento registrato dal sistema ERP esista e viceversa).	Tabelle nei sistemi ERP, *event log* nei sistemi CRM, *log* delle transazioni di sistemi basati sullo scambio di messaggi, *event log* di sistemi *high-tech*. ecc.
★★	Gli eventi vengono registrati automaticamente come prodotto secondario di un sistema informativo. La copertura rispetto alla completezza è variabile, nel senso che non viene seguito un approccio sistematico per decidere quali eventi vengono tracciati. Inoltre, è possibile aggirare il sistema informativo. Di conseguenza, alcuni eventi potrebbero mancare o non essere tracciati correttamente.	*Event log* prodotti da sistemi di gestione documentale e di gestione dei prodotti, *log* degli errori nell'ambito di sistemi integrati, fogli di lavoro in ambito ingegneristico, ecc.
★	Livello più basso: l'*event log* è di bassa qualità. Gli eventi tracciati possono non collimare con la realtà, e può accadere che alcuni eventi siano assenti nel *log*. I *log* che contengono eventi registrati manualmente presentano tipicamente queste caratteristiche.	Tracce lasciate nei documenti cartacei che circolano all'interno di un'organizzazione (come note a margine e annotazioni), *record* medici cartacei, ecc.

Tabella 1: Livelli di maturità di un *event log*.

relazione stabilita durante la fase 2, tra *log* e modello, può essere usata per estendere il modello stesso (ad esempio, i *timestamp* di eventi associati si possono essere utilizzati per stimare i tempi di attesa delle attività). Ciò può essere usato per rispondere ad ulteriori domande e può far scaturire nuove azioni. Infine, i modelli costruiti durante la fase 3 si possono utilizzare per il supporto alle decisioni (fase 4). La conoscenza estratta a partire dai dati storici sugli eventi è combinata con informazioni circa le istanze in esecuzione. In questo modo è possibile modificare, predire e raccomandare. È opportuno evidenziare che le fasi 3 e 4 possono essere eseguite solo se il processo è sufficientemente stabile e strutturato.

Attualmente, esistono tecniche e strumenti che permettono di realizzare tutte le fasi riportate in Fig. 5. Tuttavia il *Process Mining* è un paradigma relativamente nuovo e la maggior parte degli strumenti disponibili non sono maturi. Inoltre, i nuovi utenti spesso non sono a conoscenza del potenziale e delle limitazioni del *Process Mining*. Per tali ragioni, questo manifesto fornisce alcuni principi guida (cfr. Sezione 3) e nuove sfide (cfr. Sezione 4) per potenziali utenti e per ricercatori e sviluppatori, interessati a far avanzare lo stato dell'arte in questa disciplina.

3. Principi Guida

Come per ogni nuova tecnologia, ci sono degli errori ovvi che possono essere commessi nell'applicazione di tecniche di *Process Mining* su casi di studio reali. A tal proposito, vengono di seguito enumerati sei *principi guida* per aiutare utenti e analisti nella prevenzione di tali errori.

PG1: gli eventi devono essere trattati come entità di prima classe

Il punto di partenza di ogni attività di *Process Mining* sono gli eventi. Indichiamo una collezione di eventi con il termine *log degli eventi* (*event log*), ma questo non significa che gli eventi debbano necessariamente essere memorizzati all'interno di file di *log* dedicati. Gli eventi possono essere ad esempio memorizzati all'interno di tabelle relazionali, *log* di messaggi, archivi di posta elettronica, *log* contenenti transazioni, ed altre sorgenti

informative. Più importante, rispetto al formato di memorizzazione, è la *qualità* dei *log*. La qualità di un risultato di una tecnica *Process Mining* è strettamente legata alla qualità dei dati di ingresso. Di conseguenza, i *log* devono essere trattati come *entità di prima classe* nell'ambito dei sistemi informativi che supportano i processi da analizzare. Sfortunatamente, i *log* sono spesso un mero "prodotto secondario" utilizzato a fini di profilazione e *debug*. Per esempio, i dispositivi medici di *Philips Healthcare* registrano eventi semplicemente perchè gli sviluppatori hanno introdotto delle "istruzioni di stampa" all'interno del codice. Anche se esistono alcune linee guida informali per l'aggiunta di queste istruzioni al codice, è necessario un approccio più sistematico per migliorare la qualità degli *event log*. I dati legati agli eventi devono essere considerati come entità di prima classe (piuttosto che come entità secondarie).

Ci sono diversi criteri per valutare la qualità di tali dati. Gli eventi dovrebbero essere *attendibili*, ovvero permettere di poter assumere con certezza che gli eventi registrati siano effettivamente accaduti e che gli attributi degli eventi siano corretti. Gli *event log* dovrebbero essere *completi*, ovvero, una volta fissato un contesto preciso, nessun evento dovrebbe mancare. Ogni evento registrato dovrebbe avere una *semantica* ben definita. Inoltre, i dati legati agli eventi dovrebbero essere *sicuri*, ovvero registrati rispettando criteri di *privacy* e sicurezza. Ad esempio, gli attori coinvolti dovrebbero avere la consapevolezza di quali sono i tipi di evento che vengono registrati e come queste informazioni vengono utilizzate.

La Tabella 1 definisce cinque livelli di qualità dei *log* che variano da qualità eccellente (*****) a qualità povera (*). Ad esempio, gli *event log* di *Philips Healthcare* si pongono a livello ***, ovvero vengono tracciati automaticamente e il comportamento registrato corrisponde alla realtà, ma non viene seguito un approccio sistematico nell'assegnare un significato agli eventi e nel garantire copertura ad un certo livello. Le tecniche di *Process Mining* sono applicabili su *log* di livello *****, **** e *** .

In linea di principio, è possibile applicare tecniche di *Process Mining* anche utilizzando *log* di livello ** o *.

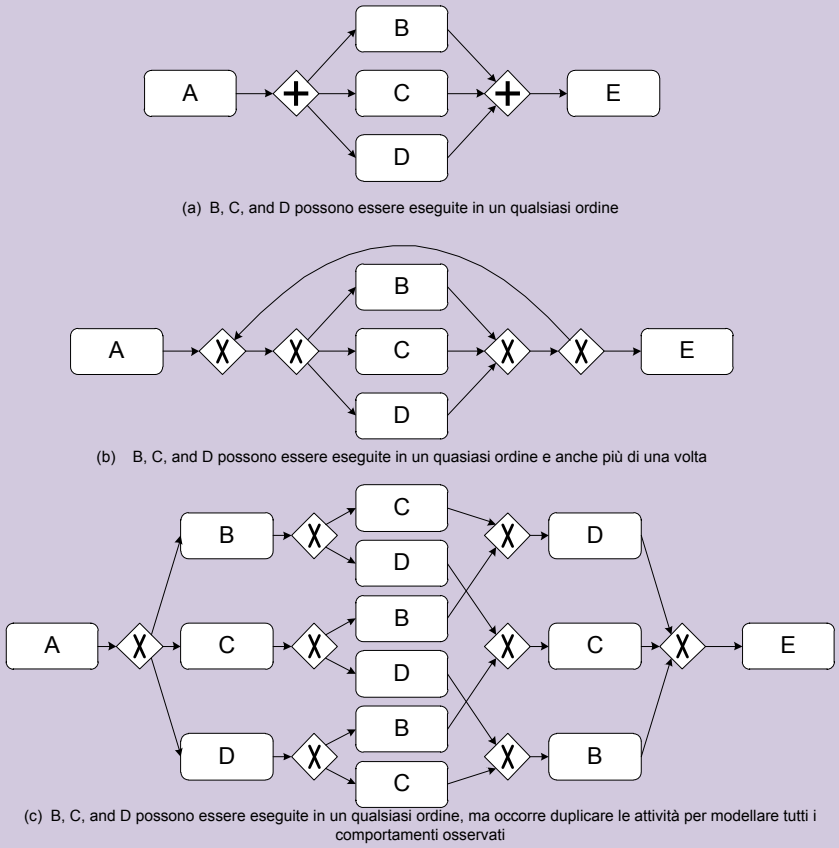


Figura 6: Esempio che illustra le problematiche che emergono quando la concorrenza (ovvero *AND-split/join*) non può essere espressa direttamente. Nell'esempio solo tre attività (B, C, e D) sono in parallelo. Si immaginino i modelli di processo ottenuti quando ci sono 10 attività concorrenti (2^{10}=1024 stati e 10! = 3,628,800 possibili sequenze di esecuzione).

L'analisi di questi *log* è però tipicamente problematica e i risultati ottenuti non sono affidabili. Di fatto, non è significativo applicare tecniche di *Process Mining* su *log* di livello *.

Al fine di beneficiare del *Process Mining*, le aziende dovrebbero puntare a tracciare *event log* al più elevato livello di qualità possibile.

PG2: L'estrazione dei log deve essere guidata da domande

Come mostrato in Fig. 5, le attività di *Process Mining* richiedono di essere guidate da domande. Senza domande concrete è molto difficile estrarre dati ed eventi significativi. Si considerino, ad esempio, le migliaia di tabelle presenti nel *database* di un sistema ERP come SAP. Senza domande concrete è impossibile selezionare quali siano le tabelle rilevanti per l'estrazione dei dati.

Un modello di processo come quello descritto in Fig. 1 descrive il ciclo di vita delle istanze di processo (*case*) di una certa tipologia. Di conseguenza, prima di applicare una qualunque tecnica di *Process Mining*, è necessario scegliere la tipologia di *case* da analizzare. Questa scelta deve essere guidata da domande specifiche a cui bisogna fornire risposta e questo può non essere per nulla banale. Si consideri, ad esempio, la gestione degli ordini di clienti. Ogni ordine può essere costituito da più linee d'ordine poiché il cliente ha la possibilità di richiedere più prodotti all'interno dello stesso ordine. Un ordine del cliente può quindi dare luogo a molteplici spedizioni. Una spedizione può riferirsi a più linee d'ordine di ordini distinti. Di conseguenza, c'è una corrispondenza molti a molti tra ordini e spedizioni e

una relazione uno a molti tra ordini e linee d'ordine. Dato un *database* con i dati legati ad ordini, linee d'ordine, e spedizioni, ci sono diversi modelli di processo che possono essere derivati. Si può decidere di estrarre i dati con l'obiettivo di descrivere il ciclo di vita dei singoli ordini. D'altro canto, è anche possibile estrarre i dati allo scopo di inferire il ciclo di vita delle singole linee d'ordine o delle singole spedizioni.

PG3: Occorre supportare concorrenza, punti di scelta e altri costrutti di base legati al flusso di controllo

Esistono molti linguaggi diversi per la modellazione dei processi (ad esempio BPMN, EPC, reti di Petri, BPEL, e diagrammi UML delle attività). Alcuni di questi linguaggi comprendono molti costrutti di modellazione (ad esempio, BPMN offre più di 50 elementi grafici distinti) mentre altri sono molto più minimali (ad esempio, le reti di Petri sono costituite solamente da tre elementi distinti: posti, transizioni, e archi).

La descrizione del flusso di controllo costituisce la struttura portante di ogni modello di processo. I costrutti di base per la modellazione del flusso di controllo (chiamati anche *workflow pattern*) supportati da tutti i linguaggi convenzionali sono la sequenza, il parallelismo (*AND-split/join*), i punti di scelta (*XOR-split/join*) e i cicli. Chiaramente, questi *pattern* devono essere supportati dalle tecniche di *Process Mining*. Purtroppo, alcune tecniche non sono capaci di gestire la concorrenza e supportano unicamente catene di *Markov*/sistemi a transizione.

La Fig. 6 mostra le implicazioni dell'uso di tecniche di *Process Mining* che non sono capaci di scoprire la concorrenza (ovvero di inferire *AND-split/join*). Si consideri un log L={ ⟨A, B, C, D, E⟩ , ⟨A, B, D, C, E⟩ , ⟨A, C, B, D, E⟩ , ⟨A, C, D, B, E⟩ , ⟨A, D, B, C, E⟩ , ⟨A, D, C, B, E⟩ }. L contiene istanze che iniziano con A e terminano con E. Le attività B, C, e D sono ordinate in tutti i possibili modi tra A ed E. Il modello BPMN in Fig. 6(a) mostra una rappresentazione compatta del processo utilizzando due costrutti di parallelismo. Si supponga che la tecnica di *Process Mining* impiegata non supporti questi costrutti. In tal caso, gli altri due modelli BPMN in Fig. 6 sono ovvi candidati. Il modello BPMN in Fig. 6(b) è compatto ma supporta troppi comportamenti (ad esempio, istanze come ⟨A, B, B, B, E ⟩ ma codifica tutte le sequenze in maniera esplicita, e non rappresenta quindi il log in maniera compatta. L'esempio mostra che per modelli reali costituiti da molte attività potenzialmente concorrenti, quando la concorrenza non è supportata i modelli ottenuti soffrono di *underfitting* (ovvero supportano troppi comportamenti) e/o sono estremamente complessi.

Come mostrato in Fig. 6, è importante supportare almeno i *workflow pattern* di base. Inoltre, al di là dei *pattern* di base è desiderabile supportare anche gli *OR-split/join*, poiché permettono di dare una rappresentazione compatta alle decisioni inclusive e ai punti di sincronizzazione parziale.

PG4: gli eventi devono essere legati ad elementi del modello

Come indicato nella Sezione 2, è un errore diffuso quello di considerare il *Process Mining* come limitato al *discovery* del flusso di controllo. Come mostrato in Fig. 1, il modello di processo estratto può coprire vari aspetti (organizzativo, temporale, legato ai dati, ecc.). Inoltre, il *discovery* è solamente una delle tre diverse tipologie di *Process Mining* mostrate in Fig. 3. Le altre due tipologie di *Process Mining* (*conformance checking* e *enhancement*) sono fortemente basate sulla corrispondenza tra gli *elementi nel modello* e gli *eventi nel log*. Questa corrispondenza può essere esplorata per fare un "*replay*" del *log* sul modello. Questo *replay* può essere impiegato per rilevare discrepanze tra *event log* e modello, ad esempio alcuni eventi contenuti nel *log* potrebbero non essere possibili secondo il modello. Le tecniche di *conformance checking* quantificano e rilevano tali discrepanze. I *timestamp* contenuti nel *log* possono essere utilizzati per analizzare l'andamento temporale durante il *replay*. Le differenze temporali tra attività correlate causalmente possono essere utilizzate per aggiungere al modello i tempi di attesa stimati. Questi esempi mostrano che la relazione tra gli eventi nel *log* e gli elementi nel modello rappresenta il punto di partenza per diversi tipi di analisi.

In alcuni casi stabilire questa corrispondenza può non essere per nulla banale. Ad esempio, un evento potrebbe riferirsi a due attività differenti oppure tale collegamento potrebbe non essere chiaro. Queste ambiguità devono essere rimosse al fine di interpretare correttamente i risultati del *Process Mining*. Al di là del problema di collegare gli eventi alle attività, c'è anche la questione di legare gli eventi alle istanze di processo. Questa problematica è generalmente nota con il nome di *event correlation*.

PG5: i modelli devono essere trattati come astrazioni utili della realtà

I modelli derivati dagli eventi forniscono *viste della realtà*. Ogni vista costituisce una astrazione utile del comportamento catturato nel *log*. Dato un *log* degli eventi, esistono molteplici viste utili. Inoltre, i vari *stakeholder* possono richiedere viste diverse. Di fatto, i modelli derivati dai *log* possono essere interpretati come "mappe" (geografiche).

Questo principio guida ha delle importanti implicazioni, due delle quali sono descritte di seguito. Innanzitutto, è importante notare che non esiste "la mappa" per una particolare area geografica. Dipendentemente dall'uso che se ne vuole fare, esistono mappe diverse: cartine stradali, mappe dei sentieri, delle piste ciclabili, ecc. Tutte queste mappe mostrano una vista della medesima realtà e sarebbe impensabile assumere l'esistenza di una "mappa perfetta". Lo stesso discorso vale per i modelli di processo: il modello dovrebbe enfatizzare gli aspetti rilevanti per una certa tipologia di utenti. I modelli derivati possono focalizzarsi su aspetti diversi (flusso di controllo, flusso dei dati, tempi, risorse, costi, ecc.) e mostrarli a differenti livelli di granularità e precisione. Ad esempio, un *manager* potrebbe richiedere di visualizzare un modello di processo "grezzo" centrato sui costi, mentre un analista potrebbe volere un modello di processo molto dettagliato

focalizzato sulle deviazioni rispetto al flusso di lavoro normale. Si noti anche che *stakeholder* diversi potrebbero necessitare di viste del processo su diversi livelli: *livello strategico* (decisioni effettuate a questo livello hanno effetti a lungo termine e sono basate su dati ed eventi aggregati rispetto a un intervallo di tempo ampio), *livello tattico* (le decisioni a questo livello hanno effetti a medio termine e sono principalmente basate su dati recenti), e *livello operativo* (le scelte a questo livello hanno effetto immediato e si basano sulle informazioni legate alle istanze di processo attualmente in esecuzione).

Inoltre, è utile adottare la metafora della cartografia quando si tratta di produrre mappe comprensibili. Ad esempio, le cartine stradali astraggono da città e strade poco rilevanti. Gli aspetti meno significativi sono tralasciati oppure dinamicamente raggruppati in forme aggregate (ad esempio, le strade e i sobborghi si amalgamano nelle città). I cartografi non solo eliminano i dettagli irrilevanti, ma utilizzano anche i colori per sottolineare le caratteristiche importanti. Inoltre, gli elementi grafici hanno una dimensione che riflette quanto sono importanti (ad esempio, la larghezza delle linee e la dimensione dei punti può variare). Le mappe geografiche fissano anche un'interpretazione chiara degli assi x e y, ovvero la distribuzione degli elementi di una mappa non è arbitraria perché le loro coordinate hanno un preciso significato. Tutte queste osservazioni contrastano profondamente con i modelli di processo tradizionali, i quali tipicamente non fanno uso di colori, dimensioni e disposizione spaziale per rendere i modelli di più facile lettura. D'altro canto, le idee impiegate nella cartografia possono essere facilmente incorporate nella costruzione delle mappe di processo inferite. Ad esempio, la dimensione di un'attività può essere utilizzata per riflettere la propria frequenza o un'altra proprietà legata alla sua importanza (ad esempio, impiego di risorse o costi). La larghezza di un arco può rappresentare l'importanza della dipendenza causale corrispondente, e il suo colore può essere utilizzato per evidenziare colli di bottiglia. Questo discorso mostra che è fondamentale scegliere la giusta rappresentazione e tararla per i suoi utilizzatori finali. Questo è importante per mostrare i risultati agli utenti e per guidare gli algoritmi di *discovery* verso modelli adeguati (si veda anche la sfida S5).

PG6: il Process Mining deve essere un processo continuo

Il *Process Mining* può aiutare nella creazione di "mappe" significative, direttamente connesse con i dati. Sia dati storici che attuali possono essere proiettati su questi modelli. Inoltre, i processi cambiano mentre vengono analizzati. Data la natura dinamica dei processi, non è consigliabile vedere il *Process Mining* come un'attività eseguita una volta sola. L'obiettivo non dovrebbe essere quello di creare un modello fisso, ma di infondere vita nei modelli di processo così da incoraggiare utenti e analisti nell'osservarli giornalmente.

Si compari ciò con l'uso di *mashup* che impiegano la georeferenziazione. Ci sono migliaia di *mashup* che sfruttano Google Maps (ad esempio, applicazioni che proiettano su mappe selezionate informazioni riguardanti le condizioni del traffico, gli immobili in vendita, i ristoranti e *fastfood*, o le proiezioni cinematografiche). Le persone possono agilmente effettuare zoom avanti e indietro sulle mappe e interagire con esse (ad esempio, dei semafori vengono inseriti nella mappa e l'utente può selezionare un problema specifico per ottenere dettagli su esso). Dovrebbe essere possibile fare *Process Mining* su eventi in tempo reale. Sfruttando la "metafora delle mappe", possiamo pensare ad eventi dotati di coordinate GPS che vengono proiettati su mappe a tempo di esecuzione. Analogamente a un sistema di navigazione automobilistica, gli strumenti di *Process Mining* possono aiutare gli utenti (a) guidandoli nella navigazione attraverso i processi, (b) proiettando informazioni dinamiche sulle mappe di processo (ad esempio mostrando "semafori" nell'ambito di un processo aziendale), e (c) effettuando predizioni sulle istanze di processo attive (ad esempio stimando il "tempo di arrivo" di un'istanza in ritardo). Questi esempi mostrano che i modelli di processo dovrebbero essere usati più attivamente. In quest'ottica, il *Process Mining* deve essere considerato come un processo continuo capace di fornire informazioni utili operativamente su diverse scale temporali (minuti, ore, giorni, settimane e mesi).

Sfide:

S1: Scoperta, Fusione e Pulizia di dati di eventi

S2: Manipolare log complessi e con caratteristiche diverse

S3: Creare Representativi Benchmarks

S4: Trattare con Concept Drift

S5: Migliorare i limiti di rappresentazione nel Process Discovery

S6: Valutare tra i Criteri di Qualità Fitness, Simplicity, Precision, e Generalization

S7: Cross-Organizational Mining

S8: Supporto alle decisioni

S9: Combinare il Process Mining con altri tipi di analisi

S10: Migliorare l'usabilità per gli utenti non esperti

S11: Migliorare la comprensibilità per gli utenti non esperti

4. Sfide

Il *Process Mining* è un importante strumento per le moderne organizzazioni che devono gestire processi operativi complessi. Da un lato c'è l'incredibile crescita della quantità di dati disponibili, dall'altro la necessità di allineare processi e informazioni in modo da soddisfare i requisiti di conformità, efficienza e di servizio ai clienti. Sebbene il *Process Mining* sia già utilizzato, ci sono ancora importanti sfide da affrontare; ciò evidenzia il fatto che il *Process Mining* è ancora una disciplina emergente. Di seguito, verranno elencate alcune di queste sfide. La lista naturalmente non è completa e, nel tempo, potrebbe allungarsi con nuove sfide o ridursi grazie al superamento di quelle attuali.

S1: Scoperta, Fusione e Pulizia di dati di eventi

L'estrazione di dati da analizzare con le tecniche di *Process Mining* richiede ancora notevole quantità di lavoro. In generale, occorre superare diversi ostacoli:

- I dati possono essere *distribuiti* su diverse sorgenti informative e c'è, quindi, la necessità di fondere queste informazioni. Ciò diventa particolarmente problematico se si usano diversi identificatori in altrettante sorgenti informative. Ad esempio, per identificare una persona un sistema può usare il nome e la data di nascita, mentre un altro sistema usa il codice fiscale.
- I dati sono spesso "orientati agli oggetti" piuttosto che "orientati ai processi". Per esempio, prodotti, carrelli e contenitori possono avere un RFID *tag* associato e gli eventi memorizzati riferirsi a questo *tag*. Tuttavia, per poter monitorare uno specifico ordine d'acquisto, tutti questi eventi orientati agli oggetti, dovranno essere fusi e preprocessati.
- I dati possono essere *incompleti*. Un problema tipico è che gli eventi memorizzati, non sono esplicitamente legati ad una istanza di processo. Spesso è possibile derivare questa informazione, ma ciò può richiedere una considerevole quantità di lavoro. Anche il riferimento temporale può mancare in alcuni eventi. Può essere allora necessario interpolare *timestamp* in modo da ottenere le informazioni temporali.
- L'*event log* può contenere *outliers*, cioè, comportamenti inusuali spesso anche definiti *rumore*. Come definire gli *outliers*? Come scoprirli? Queste domande devono trovare risposta per poter avere dati puliti.
- Il *log* può contenere eventi a *differente livello di granularità*. In un *event log* di un sistema informativo ospedaliero, gli eventi possono riferirsi ad un semplice esame del sangue o a complesse procedure chirurgiche. Anche i *timestamp* hanno differenti livelli di granularità che va da una precisione in millisecondi (28-9-2011:h11m28s32ms342) ad una meno dettagliata, per esempio giorni (28-9-2011).
- Gli eventi si riferiscono ad un particolare *contesto* (tempo, carico di lavoro, giorno della settimana, ecc.). Il contesto può spiegare alcuni fenomeni; ad esempio il tempo di risposta è più lungo di quello usuale se il lavoro è in corso di svolgimento oppure se l'esecutore è in vacanza. Per l'analisi dei *log*, è necessario tenere conto del contesto. Ciò implica che i singoli eventi vengano fusi con i dati di contesto. In tal caso, si presenta un "problema di dimensionalità" per cui l'analisi diventa intrattabile quando si aggiungono troppe variabili.

Per affrontare i problemi appena elencati, è necessario disporre di strumenti e metodologie migliori. Inoltre, come già ricordato in precedenza, le organizzazioni devono considerare gli *event log* come un'entità di prima classe piuttosto che un prodotto collaterale. L'obiettivo è di ottenere *log* di livello (vedi Tabella 1). A tal proposito, i risultati ottenuti nel campo del *data warehousing* sono utili per garantire un'alta qualità degli *event log*. Per esempio, una semplice verifica durante la fase di caricamento dei dati, aiuta a ridurre significativamente la quantità di dati errati.

S2: Manipolare log complessi e con caratteristiche diverse

I *log* possono avere caratteristiche molto differenti. Alcuni possono essere di grandi dimensioni, il che li rende molto difficili da trattare, altri possono essere così piccoli da non essere sufficienti per trarne alcuna informazione.

In alcuni domini, la quantità di eventi memorizzati è talmente grande che è necessario ulteriore lavoro per migliorare le performance e la scalabilità delle tecniche che li manipolano. Per esempio, ASML deve monitorare continuamente la produzione dei suoi *scanner*. Si tratta di dispositivi usati da varie organizzazioni (per esempio Samsung e Texas Instruments) nella produzione di *chip* (approssimativamente il 70% dei *chip* esistenti sono prodotti con scanner di ASML). In domini come questo, gli strumenti esistenti hanno notevoli difficoltà a gestire dati dell'ordine di *petabytes*. Oltre al numero di eventi memorizzati, ci sono poi altre caratteristiche da considerare, quali ad esempio il numero medio di eventi, la similarità tra le istanze di processo, il numero di eventi unici, il numero di percorsi (nel flusso di controllo) unici. Consideriamo un *log* L! con le seguenti caratteristiche: 1000 istanze, con una media di 10 eventi per istanza, e piccole variazioni (cioè, molte istanze seguono lo stesso percorso di esecuzione, o quasi). Il *log* L2 contiene appena 100 istanze, ma ci sono in media 100 eventi per istanza e ogni istanza segue un percorso diverso (unico). Ci si aspetta che L2 sia molto più difficile da analizzare rispetto a L1 sebbene entrambi i *log* abbiano una dimensione simile (circa 10.000 eventi). Quando un *log* contiene un unico comportamento è probabile che sia incompleto. Le tecniche di *Process Mining* dovrebbero gestire l'incompletezza del *log* assumendo l'ipotesi di "mondo aperto": il fatto che qualcosa non sia avvenuto non implica che non possa avvenire in futuro. Con questa assunzione, trattare con *log* di piccole dimensioni e grande variabilità è una sfida importante. Come ricordato in precedenza, alcuni *log* contengono eventi ad un bassissimo livello di astrazione. Si tratta di *log* che sono tipicamente molto estesi e per i quali i singoli eventi memorizzati essendo di basso livello sono di scarso interesse per gli *stakeholders*. Quindi potrebbe essere necessario aggregare gli eventi di basso livello in eventi di alto livello. Per esempio, nell'analizzare il processo di diagnostica e di cura di un gruppo di pazienti, si potrebbe non essere interessati ai test individuali memorizzati nel sistema informativo del laboratorio ospedaliero. Alle organizzazioni servirebbero metodi per verificare e testare se un *event log* è adatto ad essere utilizzato per tecniche di *Process Mining*. Gli strumenti a disposizione dovrebbero consentire di verificare rapidamente se un certo *dataset* è o non è utilizzabile, cioè indicare eventuali criticità legate alle *performance* e avvertire se si tratta di *log* incompleti o troppo dettagliati.

S3: Creare representativi benchmarks

Il *Process Mining* è ancora una tecnologia emergente e ciò spiega perché non sono ancora disponibili dei *dataset* di prova (*benchmark*) di buona qualità. Infatti, ci sono molte tecniche di *process discovery* e varie società che le

commercializzano nei loro prodotti, ma non ci sono metriche per stabilire la qualità di ciascuna tecnica. Sebbene ci siano grosse differenze in termini di funzionalità e di *performance*, non è per nulla facile confrontare le varie tecniche e gli strumenti che l'implementano. Ecco perché occorre definire *benchmark* di qualità soddisfacente e criteri di qualità di riferimento. In genere si dispone di molti *benchmark* per testare tecniche di *data mining*. Proprio questi *benchmark* hanno rappresentato uno stimolo per produttori di *tool* e ricercatori affinché migliorassero le *performance* delle proprie tecniche. Al contrario, per il *Process Mining* tutto questo rappresenta una sfida molto più ardua. Per esempio, il modello relazionale introdotto da Codd nel 1969 è semplice ed ampiamente supportato. Per questo è anche piuttosto facile convertire dati da un *database* ad un altro senza avere problemi di interpretazione. Per i processi non esiste un modello così semplice. Gli *standard* proposti per modellare processi sono molto complicati e pochi produttori supportano esattamente lo stesso insieme di concetti. In altri termini i processi sono molto più complessi dei dati espressi in forma tabellare. Nonostante ciò, è altrettanto importante creare *benchmark* per il *Process Mining*. Qualcosa in questa direzione è stato fatto. Ad esempio, sono state proposte diverse metriche per misurare la qualità dei risultati di *Process Mining* (*fitness*, *simplicity*, *precision*, e *generalization*). Inoltre, diversi *event log* sono pubblici e disponibili (cf. www.processmining.org). Ne è un esempio il *log* usato per la prima edizione della competizione BPIC'11 organizzata dalla *task force* (cfr. doi: 10.4121/uuid:d9769f3d-0ab0-4fb8-803b-0d1120ffcf54). Se *benchmark* basati su *dataset* reali sono necessari, è necessario anche creare *dataset* sintetici con specifiche caratteristiche che aiutino a sviluppare tecniche di *Process Mining* in grado di gestire *log* incompleti, rumore o adatti per specifiche popolazioni di processi.

Oltre alla creazione di *benchmarks*, bisogna anche trovare un maggiore accordo sui criteri usati per stabilire la qualità dei risultati ottenuti dal *Process Mining* (si veda anche il paragrafo S6). A tal proposito è possibile adattare le tecniche di *cross-validation* già usate nel *data mining* per valutare i risultati. Si consideri ad esempio la tecnica del k-*fold checking*, in cui un *log* viene suddiviso in k parti e di queste k-1 parti sono utilizzate per apprendere il modello di processo, mentre la parte restante può essere usata per validare il risultato attraverso tecniche di *conformance*. Questa operazione è ripetuta k volte in modo da avere un risultato più affidabile.

S4: Trattare con concept drift

Il termine *concept drift* indica il fatto che il processo che si sta analizzando è in continua evoluzione. Ciò significa ad esempio, che se all'inizio di un *log* due attività sono in parallelo, da un certo punto in poi nel *log* queste stesse attività sono sequenziali. Il cambiamento di un processo può essere legato a cambiamenti periodici/stagionali (per esempio "a Dicembre c'è più richiesta" oppure "il venerdì sera ci sono pochi impiegati disponibili") oppure perché cambiano le condizioni al contorno (per esempio "il mercato diventa più competitivo"). In ogni caso i cambiamenti impattano sui processi. Per questo è di vitale importanza riuscire a individuarli ed analizzarli. Un modo per scoprire il *concept drift* in un processo è di suddividere il *log* in *log* di piccole dimensioni e analizzarli in dettaglio. Tale analisi di"secondo livello" richiede però molti più dati. Nonostante questo, essendo pochi i processi che possono considerarsi definitivamente stabilizzati, comprenderne il *concept drift* risulta di primaria importanza per la loro gestione. Concludendo, bisogna investigare maggiormente e disporre di strumenti adatti ad analizzare il *concept drift*.

S5: Migliorare i limiti di rappresentazione nel Process Discovery

Le tecniche di *process discovery* forniscono come risultato un modello di processo in un particolare linguaggio (per esempio BPMN o reti di Petri). Tuttavia, è importante separare la visualizzazione del risultato, dalla rappresentazione adottata durante il processo di *discovery*. La scelta del linguaggio *target* spesso comporta implicitamente alcune assunzioni. Limita lo spazio di ricerca: i processi che non possono essere rappresentati nel linguaggio *target* non possono essere individuati. Questo aspetto, noto anche come "*representational bias*", impone che la scelta della rappresentazione deve essere una scelta consapevole e non (solo) dettata da una preferenza per una specifica rappresentazione grafica.

Si consideri ad esempio la Fig. 6: a seconda se il linguaggio *target* supporta o meno la concorrenza, si avranno ripercussioni non solo sulla visualizzazione del modello derivato, ma anche sulla classe di modelli supportati dall'algoritmo. Infatti se la concorrenza non è supportata (Fig. 6(a) non è possibile), se non sono ammesse attività multiple con la stessa etichetta (Fig. 6(c) non è possibile), allora solo particolari modelli come quello in Fig. 6(b) sono ammessi. L'esempio mostra quindi che la scelta del *representational bias* deve necessariamente essere attenta e consapevole.

S6: Valutare tra i criteri di qualità fitness, simplicity, precision, e generalization

I *log* sono spesso incompleti perché, ad esempio, hanno memorizzato un solo tipo di comportamento. I modelli di un processo possono supportare un numero esponenziale se non infinito di tracce diverse (in caso di cicli). Inoltre alcune tracce possono essere più probabili di altre. Non è quindi realistico assumere che tutte le possibili tracce siano presenti nel *log*. Per avere un'idea di ciò, si consideri un processo di 10 attività che possono essere eseguite in parallelo e un *log* di esecuzione contenente 10.000 istanze. Il numero totale di possibili combinazioni in un modello contenente 10 attività in parallelo è 10! = 3.628.800. È dunque impossibile che ci siano tutte le combinazioni nel *log* in quanto abbiamo poche istanze memorizzate (10.000) a fronte di tutte le possibili tracce (3.628.800). Anche qualora il *log* fosse molto grande, dell'ordine di milioni di istanze, è comunque improbabile che siano state memorizzate tutte le possibili variazioni. Un'ulteriore complicazione deriva dal fatto che alcune alternative sono meno probabili di altre. È possibile in tal caso considerarle come "rumore". È ovviamente impossibile

costruire un modello significativo per questo tipo di istanze. Di fatto il modello derivato dovrebbe astrarre dal rumore. I comportamenti infrequenti invece dovrebbero essere meglio analizzati mediante il *conformance checking*.

Rumore e incompletezza sono una sfida per il *process discovery*. Di fatto, ci sono quattro criteri di qualità in competizione tra di loro: (a) *fitness*, (b) *simplicity*, (c) *precision*, e (d) *generalization*. Un modello con un buon *fitness* supporta la maggior parte dei comportamenti rilavati nei *log*. In particolare, un modello ha *fitness* massimo se tutte le istanze nel *log* possono essere riprodotte sul modello dall'inizio alla fine. Inoltre, il modello migliore per descrivere il comportamento rilevato in un *log* è quello più *semplice*, in accordo con il noto principio del *Rasoio di Occam*. Tuttavia, *fitness* e *simplicity* non sono sufficienti da soli ad indicare la qualità di un modello derivato da un *log*. Di fatto, è facile costruire una rete di Petri molto semplice ("*flower model*") capace di supportare tutte le tracce in un *log* (e qualsiasi *log* contenente lo stesso insieme di attività). Analogamente, non è conveniente avere un modello che catturi esattamente tutti i comportamenti registrati nel *log*. Si ricordi che il *log* contiene solo i comportamenti che è stato possibile memorizzare fino ad un dato istante ma molte tracce ammissibili potrebbero ancora essere memorizzate in futuro. Un modello è *preciso* se non ammette comportamenti che si discostano molto dai comportamenti riscontrati nel *log*. Chiaramente, il "*flower model*" non è preciso. Un modello che non è preciso è "*underfitting*". *Underfitting* indica che il modello sovrageneralizza il comportamento usuale nel *log* (cioè, cattura anche comportamenti molto differenti da quelli effettivamente memorizzati nel *log*). Un modello dovrebbe anche generalizzare senza limitarsi solo ai comportamenti memorizzati nei *log*. Un modello che non *generalizza* è "*overfitting*". *Overfitting* indica che il modello generato è estremamente specifico (cioè, è in linea con quel particolare *log*, ma un *log* registrato in un secondo momento per lo stesso processo potrebbe produrre un modello completamente differente).

Bilanciare *fitness*, *simplicity*, *precision* e *generalization* è una sfida. Questo è uno dei motivi per cui la maggior parte delle tecniche di *process discovery* necessitano di diversi parametri. Gli algoritmi devono essere migliorati per meglio bilanciare le quattro dimensioni. Inoltre, ciascun parametro usato dovrebbe essere comprensibile all'utente finale.

S7: Cross-organizational mining

Tradizionalmente, il *Process Mining* è applicato in una singola organizzazione. Tuttavia con l'enorme diffusione della tecnologia a servizi, dell'integrazione tra catene di fornitori, e del *cloud computing*, non è inconsueto dover analizzare *log* appartenenti a un più organizzazioni. In principio, ci sono due contesti applicativi per il *cross-organizational Process Mining*.

Prima di tutto, si può considerare un contesto collaborativo in cui diverse organizzazioni lavorano insieme sulla stessa istanza di processo. In tal caso, *cross-organizational Process Mining* è una sorta di "gioco del puzzle", in cui l'esecuzione dell'intero processo è suddiviso in diverse parti e distribuito alle varie organizzazioni coinvolte che devono cooperare per assicurarne il completamento con successo. Analizzare un *log* singolarmente per ciascuna organizzazione coinvolta non è sufficiente. Per derivare il processo globale, occorre fondere insieme tutti i *log* delle singole organizzazioni, il che non è banale visto che gli eventi devono essere correlati tra le diverse organizzazioni.

In secondo luogo, si può considerare un contesto in cui diverse organizzazioni che eseguono essenzialmente lo stesso processo, decidono di condividere esperienze, conoscenza, o una comune infrastruttura. Si consideri l'esempio di Salesforce.com. Saleforce gestisce e supporta il processo di vendita di molte organizzazioni. Da un lato, tutte queste organizzazioni condividono la stessa infrastruttura (processi, *database*, etc.). Dall'altro, però, non sono obbligate a seguire strettamente uno specifico modello di processo, poiché il sistema può essere configurato in modo da supportare varianti dello stesso processo. Un ulteriore esempio è il caso di processi eseguiti nelle amministrazioni pubbliche (per esempio un ufficio che rilascia permessi). Sebbene tutte le amministrazioni di ciascuna regione devono supportare un certo numero di processi comuni, ci possono essere delle differenze. Risulta interessante quindi analizzare tali differenze tra le varie organizzazioni. In questo modo queste organizzazioni possono imparare l'una organizzazioni, cosicché ciascuna organizzazione possa imparare dall'altra e i fornitori di servizi migliorare i prodotti offerti, offrendo servizi migliori grazie ai risultati ottenuti dal *cross-organizational Process Mining*.

Per entrambi i tipi di *cross-organizational Process Mining* è comunque necessario sviluppare nuove tecniche che non possono prescindere dai problemi di *privacy* e di sicurezza. Le organizzazioni potrebbero non voler condividere informazioni sia per ragioni di competitività, sia per mancanza di fiducia. Pertanto, è anche importante sviluppare tecniche di *privacy-preserving Process Mining*.

S8: Supporto alle decisioni

Originariamente, il *focus* del *Process Mining* era l'analisi di dati storici. Oggigiorno invece molte sorgenti di dati vengono modificate (quasi) in *real-time* e si ha a disposizione una potenza computazionale sufficiente ad analizzare eventi esattamente nel momento in cui avvengono. Il concetto di *Process Mining* quindi non deve essere più ristretto ad un'analisi *off-line* ma può essere esteso anche per attività di *supporto decisionale* in tempo reale. Il supporto alle decisioni prevede tre tipi di attività: attività di "rilevazione", "predizione" e "raccomandazione". Quando un'istanza di processo si discosta dal processo predefinito, ciò può essere rilevato e il sistema può generare un *alert*. Spesso si vorrebbe poter generare questa notifica non con un'analisi *off-line* ma immediatamente (per avere ancora la possibilità di modificare il corso dell'istanza di processo). I dati storici possono essere usati per costruire modelli predittivi. Questi possono essere usati per "guidare" le istanze di processo in esecuzione. Per esempio, è possibile prevedere il tempo di esecuzione rimanente per un'istanza di processo. Basandosi su tali previsioni è anche possibile costruire sistemi capaci di fare raccomandazioni che propongono

particolari azioni da intraprendere per ridurre i costi o per diminuire i tempi di esecuzione. L'applicazione di tecniche di *Process Mining* per un'analisi *on-line* apre ulteriori sfide in termini di potenza computazionale e qualità dei dati.

S9: Combinare il Process Mining con altri tipi di analisi

Il *management* operativo (e in particolare la ricerca operativa) è una branca dell'ingegneria gestionale che si basa sulla modellazione. In questo campo viene utilizzata una grande varietà di modelli matematici che vanno dalla programmazione lineare, al *project planning*, ai modelli a coda, alle catene di Markov, alla simulazione. Il *data mining* può essere definito come "l'analisi di *dataset* (spesso molto estesi) per identificare nuove relazioni tra i dati e per descriverli con rappresentazioni che siano comprensibili e utili per chi li gestisce". A questo scopo è stata sviluppata una grande varietà di tecniche e in particolare tecniche di: classificazione (per esempio apprendimento di alberi di decisione), regressione (per esempio *k-means clustering*) e *pattern discovery* (per esempio apprendimento di regole associative).

Entrambi i campi (*management* operativo e *data mining*) forniscono importanti tecniche di analisi. La nuova sfida consiste nel combinare le tecniche messe a disposizione in questi campi con il *Process Mining*. Consideriamo, per esempio, la simulazione. Le tecniche di *Process Mining* possono essere usate per derivare un modello di simulazione basato su dati storici. Di conseguenza, il modello di simulazione può essere usato per fornire supporto operativo. Data la stretta connessione tra *log* e modello, quest'ultimo può quindi essere usato per il *replay* del processo. In particolare, è possibile eseguire simulazioni a partire dallo stato corrente del processo per fornire un "rapido salto" nel futuro basato su dati a *runtime*.

Allo stesso modo, è interessante combinare il *Process Mining* con tecniche di *analisi visuale* (in inglese *Visual Analytics*). L'analisi visuale combina l'analisi automatica con visualizzazioni interattive per una migliore comprensione di *dataset* ampi e complessi. L'analisi visuale si basa sulla sofisticata capacità di un essere umano di individuare *pattern* in dati non strutturati. Combinando le tecniche di *Process Mining* automatico con l'analisi visuale interattiva, è possibile estrarre una quantità maggiore di informazioni dai dati.

S10: Migliorare l'usabilità per gli utenti non esperti

Uno degli obiettivi del *Process Mining* è creare "modelli di processo viventi", cioè modelli di processo che possono essere usati ogni giorno piuttosto che modelli statici dimenticati in un archivio. Dati sempre nuovi possano essere usati per identificare nuovi comportamenti. Il legame tra dati e modelli di processo permette la proiezione dello stato corrente del processo e delle attività recenti su modelli continuamente aggiornati. Quindi, gli utenti possono interagire con i risultati del *Process Mining* ogni giorno. Queste interazioni da un lato sono molto importanti ma dall'altro richiedono interfacce utenti intuitive. La nuova sfida è quindi rendere trasparenti per l'utente i sofisticati algoritmi di *Process Mining* attraverso interfacce "*user-friendly*" che individuano automaticamente i parametri degli algoritmi di *mining* e suggeriscono i tipi di analisi più adatti alle diverse situazioni.

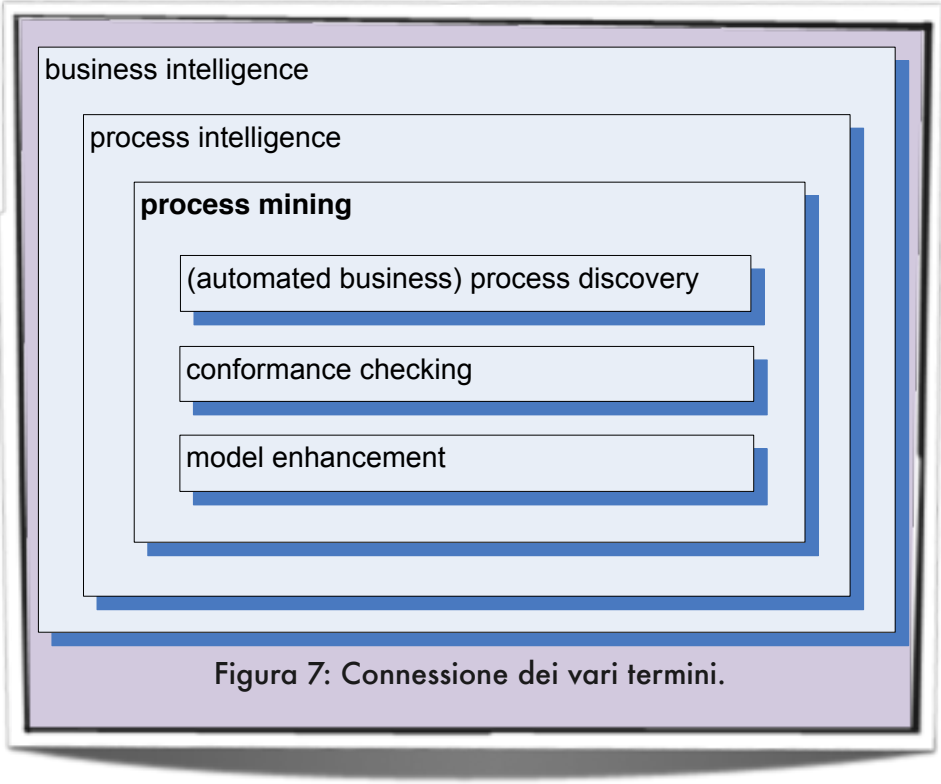


Figura 7: Connessione dei vari termini.

S11: Migliorare la comprensibilità per gli utenti non esperti

Anche se è facile generare risultati attraverso il *Process Mining*, questo non significa che tutti i risultati sono di fatto utili. L'utente può avere problemi a capire gli *output* di un algoritmo di *Process Mining* o può essere indotto a trarre conclusioni sbagliate. Per evitare questi problemi, i risultati dovrebbero essere presentati usando rappresentazioni opportune (cfr. PG5). Inoltre, l'affidabilità dei risultati dovrebbe essere sempre chiaramente specificata. In alcuni casi per esempio ci possono essere troppo pochi dati per giustificare determinate conclusioni. Di fatto, le tecniche esistenti per il *process discovery* tipicamente non avvisano l'utente quando il modello generato ha un *fitness* basso o, al contrario, è "*overfitting*". Tali tecniche producono sempre un modello anche quando risulta evidente che ci sono troppo pochi dati per giustificare qualsiasi conclusione.

La versione originale e apparsa sui proceedings dei Workshops BPM 2011, Lecture Notes in Business Information Processing, Vol.99, Springer-Verlag, 2011, ed è stata tradotta in diverse lingue. Per maggiori dettagli si veda il sito della IEEE Task Force sul Process Mining: http://www.win.tue.nl/ieeetfpm/.

Epilogo

L'*IEEE Task Force* sul *Process Mining* ha come obiettivi (a) promuovere l'applicazione del *Process Mining* (b) guidare gli sviluppatori, i consulenti, i *manager* e gli utenti nell'uso delle tecniche di *Process Mining* esistenti, e (c) stimolare la ricerca nel *Process Mining*. Questo manifesto definisce i principi e le intenzioni della *task force*. Dopo aver introdotto il *Process Mining*, il manifesto descrive una lista di alcuni principi guida (Sezione 3) e di nuove sfide (Sezione 4). I principi guida possono essere usati per evitare di compiere errori banali nell'utilizzo di queste tecniche. La lista delle nuove sfide serve per orientare il lavoro di ricerca e di sviluppo. Entrambi hanno lo scopo di incrementare il livello di maturità del *Process Mining*.

Per concludere, qualche parola sulla terminologia. Nell'ambito del *Process Mining* vengono usati i termini: *workflow mining*, (*business*) *Process Mining*, *automated business process discovery* o *business process intelligence*. Varie organizzazioni usano termini diversi per indicare gli stessi concetti. Per esempio, la Gartner usa il termine "*Automated Business Process Discovery*" (ABPD) e la Software AG usa il termine "*Process Intelligence*" per riferirsi alle loro piattaforme. Il termine "*workflow mining*" risulta invece in genere meno adatto per indicare il *Process Mining* in quanto la creazione di modelli di *workflow* è soltanto una delle tante possibili applicazioni del *Process Mining*. Allo stesso modo, l'aggiunta del termine "*business*" serve a considerare solo alcuni tipi di applicazioni: ci sono numerose applicazioni del *Process Mining* (per esempio l'analisi di sistemi ad alta tecnologia o l'analisi di siti *web*) dove il termine *business* risulta inappropriato. Sebbene il *process discovery* è un importante componente dello spettro delle tecniche di *Process Mining*, esso è soltanto uno dei possibili tipi. Il *conformance checking*, le predizioni, il *mining* organizzativo, l'analisi delle *social network* etc. rappresentano altri tipi di *Process Mining* che vanno oltre il *process discovery*.

La Fig. 7 elenca alcuni dei termini appena citati. Tutte le tecnologie e i metodi che hanno l'obiettivo di fornire informazione che può essere usata per il supporto decisionale possono essere considerate come parte della *Business Intelligence* (BI). La (*business*) *process intelligence* può essere vista come la combinazione di BI e BPM, cioè le tecniche di BI possono essere utilizzate per analizzare e migliorare i processi e la loro gestione. Il *Process Mining* può essere visto come una concretizzazione della "*process intelligence*" che considera i *log* come punto di partenza. L'*automated business process discovery* è solo uno dei tre tipi esistenti di *Process Mining*. La Fig. 7 potrebbe essere fuorviante nel senso che la maggior parte dei *tool* per BI non forniscono funzionalità di *Process Mining* così come sono state descritte in questo documento. Il termine BI viene utilizzato impropriamente per indicare uno specifico *tool* o un metodo che possono riferirsi solo ad una piccola parte del ben più ampio spettro delle tecniche di BI. Ci possono essere ragioni commerciali per usare termini alternativi. Alcuni produttori possono per esempio voler enfatizzare un aspetto particolare (per esempio *discovery* o *intelligence*). Tuttavia, per evitare ambiguità, è preferibile usare il termine *Process Mining* per la disciplina descritta in questo manifesto.

Authors

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
Moe Wynn

Glossario

Attività: un passo ben definito di un processo. Gli eventi possono riferirsi all'avvio, al completamento, alla cancellazione etc. di un'attività per un'istanza di processo specifica.
Automated Business Process Discovery: sinonimo di Process Discovery.
Business Intelligence (BI): vasta collezione di tool e metodi che usano dati per supportare il decision making.
Business Process Intelligence: sinonimo di Process Intelligence.
Business Process Management (BPM): disciplina che combina conoscenze nel campo dell'information technology e conoscenze nel campo dell'ingegneria gestionale e le applica su processi di business operativi.
Case: sinonimo di Istanza di Processo.
Concept Drift: fenomeno secondo il quale i processi cambiano spesso nel tempo. Il processo sotto osservazione può gradualmente (o improvvisamente) cambiare per esempio a causa di cambiamenti di stagione o di aumento della competizione così da renderne più difficile l'analisi.
Conformance Checking o Verifica di Conformità: analisi che stabilisce se ciò che accade nella realtà (come risulta dai log) è conforme al modello e viceversa. L'obiettivo è riscontrare discrepanze e misurare la loro gravità. Il conformance checking è uno dei tre tipi base di Process Mining.
Cross-Organizational Process Mining: applicazione di tecniche di Process Mining a event log provenienti da organizzazioni diverse.
Data Mining: analisi di dataset (estesi) per l'identificazione di nuove relazioni tra i dati e per descriverli in modo da fornire nuove informazioni su di essi.
Enhancement: uno dei tre tipi base di Process Mining. Un modello di processo è esteso o migliorato usando informazioni estratte dai log. Per esempio è possibile identificare colli di bottiglia riproducendo un event log su un modello di processo esaminando i timestamp.
Evento: un'azione registrata in un log come per esempio l'avvio, il completamento o la cancellazione di un'attività in un'istanza di processo specifica.
Event Log o Log degli Eventi: collezione di eventi usata come input per il Process Mining. Gli eventi non devono essere registrati necessariamente in un file di log (per esempio gli eventi possono essere sparsi su diverse tabelle in un database).
Fitness: misura per determinare quanto un dato modello supporta il comportamento rilevato nel log. Un modello ha fitness massimo se tutte le istanze nel log possono essere riprodotte sul modello dall'inizio alla fine.
Generalization: misura per determinare quanto il modello è capace di supportare comportamenti nascosti. Un modello è "overfitting" se non è in grado di generalizzare a sufficienza.
Istanza di Processo: entità gestita da un processo che viene di fatto analizzata. Gli eventi si riferiscono a istanze di processo. Esempi di istanze di processo sono: l'ordine di un cliente, una richiesta di assicurazione, una richiesta per un prestito, etc.
MXML: formato XML per lo scambio di event log. XES sostituisce MXML come nuovo formato per il Process Mining (indipendente dai tool specifici).
Operational Support o Supporto alle Decisioni: analisi on-line di dati con lo scopo di monitorare e influenzare le istanze di processo in esecuzione. Esistono tre differenti attività di supporto alle decisioni: rilevazione (genera un alert quando il comportamento osservato si discosta da quello modellato); predizione (predice comportamenti futuri sulla base di quelli passati, per es. predice il tempo rimanente di esecuzione); raccomandazione (suggerisce le azioni necessarie per realizzare un determinato obiettivo come per esempio minimizzare i costi).
Precision: misura che determina se il modello non ammette comportamenti che si discostano molto dai comportamenti riscontrati nell'event log. Un modello con un basso livello di precisione è "underfitting".
Process Discovery: uno dei tre tipi base di Process Mining. Sulla base di un event log costruisce un modello di processo. Per esempio, l'algoritmo ALPHA costruisce una rete di Petri identificando pattern in una collezione di eventi.
Process Intelligence: branca della Business Intelligence specifica per il Business Process Management.
Process Mining: tecniche, tool e metodi per derivare, monitorare e migliorare processi reali estraendo conoscenza dai log, oggi ampiamente disponibili nei sistemi (informativi).
Representational Bias: linguaggio scelto per presentare e costruire i risultati del Process Mining.
Simplicity: misura che rende operativo il Rasioio di Occam, cioè il modello migliore per descrivere il comportamento rilevato in un log è quello più semplice. La semplicità può essere quantificata in diversi modi, per esempio col numero di nodi e di archi presenti nel modello.
XES: è uno standard XML per descrivere event log, adottato dall'IEEE Task Force sul Process Mining come formato di default per scambiare event log (cfr. www.xes-standard.org).