

Misurare Processi di Business

R. Bruni¹ A. Corradini¹ G. Ferrari¹ T. Flagella²
R. Guanciale¹ **G.O. Spagnolo¹**

¹Dipartimento di Informatica, Università di Pisa

²Link.it

12 novembre 2011

Conferenza AICA 2011 - Torino

Contesto

Questa attività di ricerca è stata realizzata nell'ambito del progetto di *Ricerca sull'Usabilità delle Piattaforme Orientate ai Servizi* (RUPOS), con il parziale supporto della Regione Toscana

Business Process Management

- BPM affronta la modellazione, l'organizzazione, l'applicazione e l'ottimizzazione delle attività necessarie per raggiungere un determinato obiettivo (es. offrire un determinato servizio, oppure produrre un certo manufatto).
- In BPM, i processi vengono rappresentati attraverso formalismi grafici, permettendo di comunicare in modo non ambiguo le regole di business, e quindi discuterle o modificarle, tra gli svariati ruoli coinvolti che vanno dagli esperti del dominio di business o del settore, agli architetti software e sviluppatori.

Obiettivi

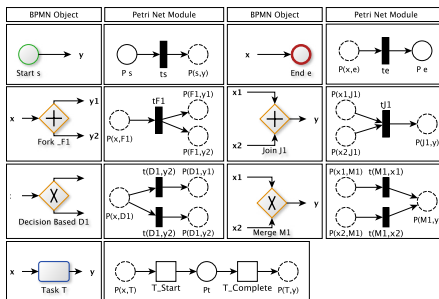
- In questo contributo ci focalizziamo su una specifica fase del BPM, che comprende il monitoraggio e la valutazione. L'obiettivo di questa fase consiste nel verificare la corretta esecuzione dei processi e misurarne le prestazioni dopo il loro deployment.

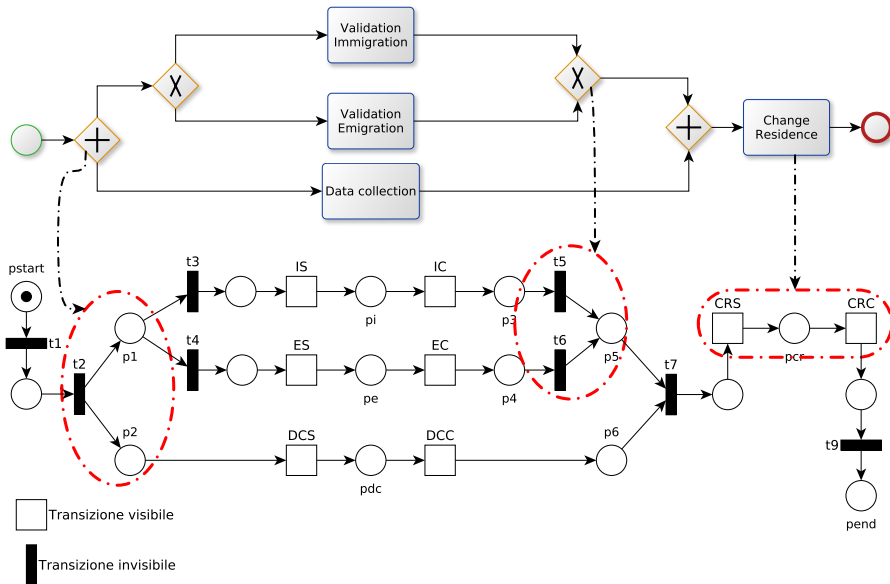
Strategia

- Adottare ed estendere esistenti metodi formali (Petri Nets)
- Integrare ed estendere esistenti infrastrutture software (ProM)
- Metodologia dei work-flow
 - ① I processi sono descritti con BPMN diagrams
 - ② Il BPMN diagram viene trasformato in una Petri Net
 - ③ I log delle istanze di processo sono processati usando tecniche disponibili per le Petri Net
 - ④ I risultati delle analisi sono proiettati indietro sul modello di partenza BPMN.

From BPMN to Petri Net

- Sfruttiamo una metodologia di trasformazione esistente (Dijkman, R.M., Dumas, M., Ouyang, C.) estesa
- Successivamente affrontiamo il problema di riportare i risultati di queste analisi sul modello BPMN di partenza.



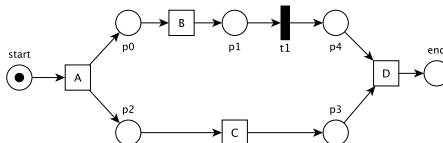


Analisi basata su Petri Nets

- Gli eventi delle istanze di processo del log sono ordinati (e.s. timestamp)
- Gli eventi sono mappati sulle transizioni della rete
- **Log Replay**: replay delle istanze di processo del log (non-blocking way)
 - ① L'algoritmo parte con un token nella piazza iniziale delle rete
 - ② Estrae dalla testa del log l'evento
 - ③ Viene effettuato il fire della corrispondente transizione
 - Se la transizione non è abilitata i token mancanti vengono creati artificialmente e chiamati **missing token**
- Metriche
 - Il numero di missing/remaining token per ogni piazza/transizione
 - Il numero di archi attraversati
 - Il tempo di soggiorno/attesa/sincronizzazione per ogni piazza.

Refined Performance example

- Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$
- Log replay transition sequence $A, B, C, t1, D$
- Resulting eager sequence $A, B, t1, C, D$

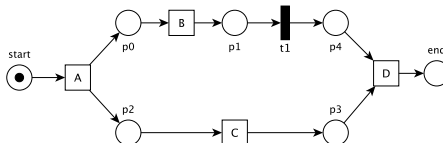


Measures



Refined Performance example

- Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$
- Log replay transition sequence $A, B, C, t1, D$
- Resulting eager sequence $A, B, t1, C, D$

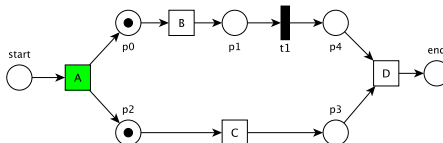


Measures



Refined Performance example

- Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$
- Log replay transition sequence $A, B, C, t1, D$
- Resulting eager sequence $A, B, t1, C, D$

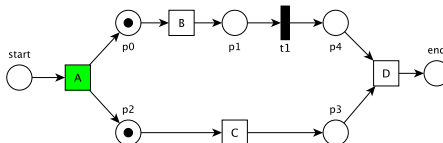


Measures

	p0	p2
×	0	0
⊗		

Refined Performance example

- Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$
- Log replay transition sequence $A, B, C, t1, D$
- Resulting eager sequence $A, B, t1, C, D$

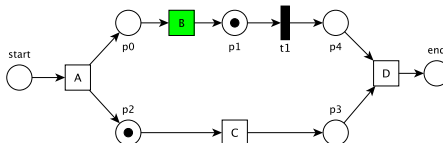


Measures

	p0	p2
⊗	0	0
⊗		

Refined Performance example

- Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$
- Log replay transition sequence $A, B, C, t1, D$
- Resulting eager sequence $A, B, t1, C, D$

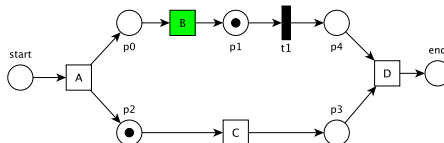


Measures

	p0	p2	p1
×	0	0	0
⊗	1		

Refined Performance example

- Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$
- Log replay transition sequence $A, B, C, t1, D$
- Resulting eager sequence $A, B, t1, C, D$

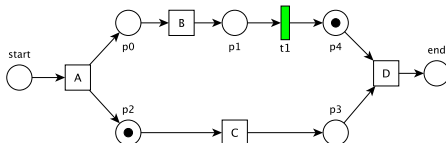


Measures

	p0	p2	p1
×	0	0	0
⊗	1		

Refined Performance example

- Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$
- Log replay transition sequence $A, B, C, t1, D$
- Resulting eager sequence $A, B, t1, C, D$

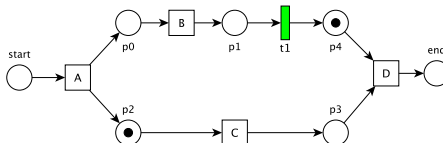


Measures

	p0	p2	p1	p3	p4
×	0	0	0		
⊗	1		0		

Refined Performance example

- Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$
- Log replay transition sequence $A, B, C, t1, D$
- Resulting eager sequence $A, B, t1, C, D$

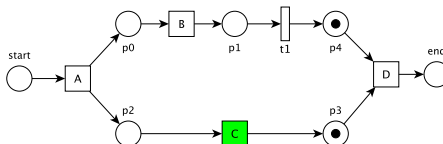


Measures

	p0	p2	p1	p3	p4
×	0	0	0		
⊗	1		0		

Refined Performance example

- Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$
- Log replay transition sequence $A, B, C, t1, D$
- Resulting eager sequence $A, B, t1, C, D$

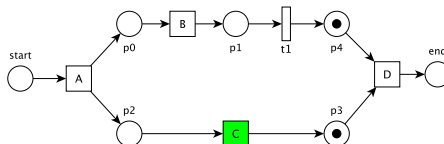


Measures

	p0	p2	p1	p3	p4
×	0	0	0	0	2
⊗	1	2	0		

Refined Performance example

- Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$
- Log replay transition sequence $A, B, C, t1, D$
- Resulting eager sequence $A, B, t1, C, D$

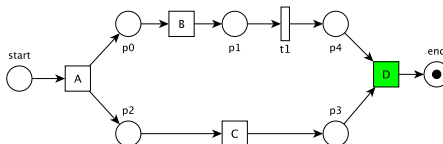


Measures

	p0	p2	p1	p3	p4
×	0	0	0	0	2
⊗	1	2	0		

Refined Performance example

- Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$
- Log replay transition sequence $A, B, C, t1, D$
- Resulting eager sequence $A, B, t1, C, D$

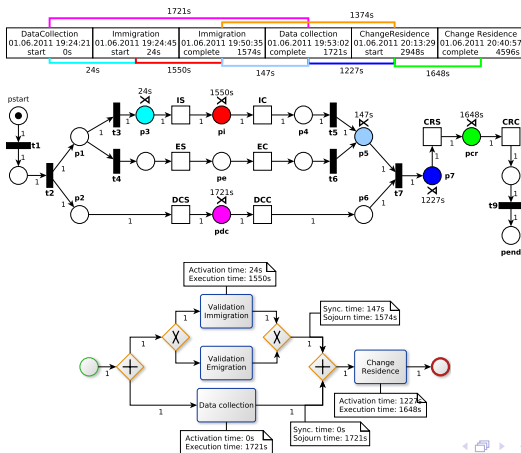


Measures

	p0	p2	p1	p3	p4
×	0	0	0	0	2
⊗	1	2	0	4	6

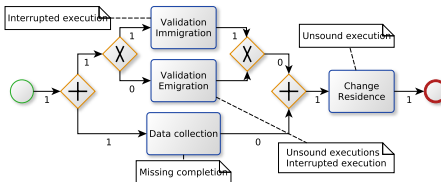
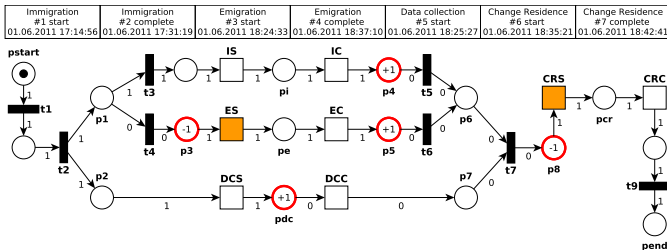
Dall'analisi al BPMN (Performance)

- **Tempi di Attesa:** sono nei place pre-set delle transizioni visibili
- **Tempi di Sincronizzazione:** sono nei place che hanno almeno una transizione nel loro post-set che dipende da un altro place.



Dall'analisi al BPMN (Conformance)

- **Missing tokens:** nei place pre-set di una transizione visibile
- **Remaining tokens** nei place post-set di una transizione visibile o di una transizione invisibile che generano più di un token.



Risultati teorici

- Affinate le tecniche per gestire le transizioni invisibili sulle Petri Net
- Proiezioni delle misure sul BPMN

Risultati di sviluppo

- Nuovi plug-ins per ProM
 - Trasformazione del modello BPMN in una Petri Net
 - Analisi di Performance e Conformance su Petri Net
 - Proiezioni delle misure di analisi sul modello BPMN di partenza.

- 1 Introduzione
 - Business Process Management
 - Sommario
- 2 Supporto alla modellazione tramite BPMN
 - From BPMN to Petri Net
- 3 Analisi basata su Petri Nets
 - Analisi basata su Petri Nets
- 4 Proiezioni dei dati sul modello
 - Dall'analisi al BPMN
- 5 Conclusioni