Misurare Processi di Business

R.Bruni A.Corradini G.Ferrari T.Flagella R.Guanciale
G.O.Spagnolo

8 novembre 2011

Conferenza aica 2011 - Torino

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

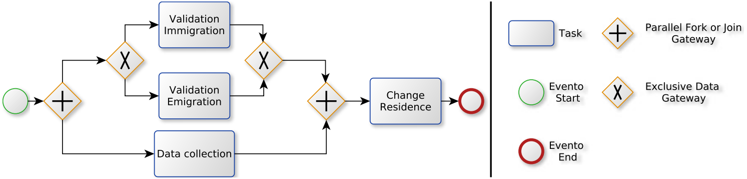
- “Group” application interactions into process instances
- Monitor process evolution
- Monitor SLAs
- Detect and inspect protocol failures

Strategia

- Adottare ed estendere esistenti metodi formali (Petri Nets)
- Integrare ed estendere esistenti infrastutture software (ProM)
- Metodologia dei work-flow
  1. I processi sono descritti con BPMN diagrams
  2. Il BPMN diagram viene trasformata 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.

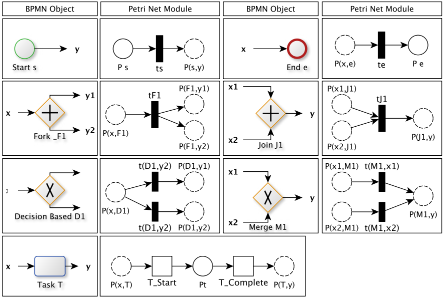
Esempio di modello BPMN

- Permette di modellare i processi ad un alto livello di astrazione, questo modello pu essere compreso o creato anche dai non addetti ai lavori.



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.

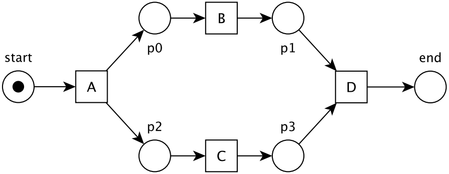


Analisi basata su Petri Nets

- 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 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.

Log-replay examples

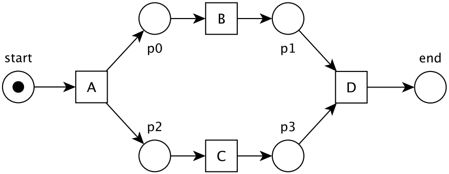
Trace log $(A, 1s), (B, 2s), (C, 4s), (D, 8s)$



Measures

Log-replay examples

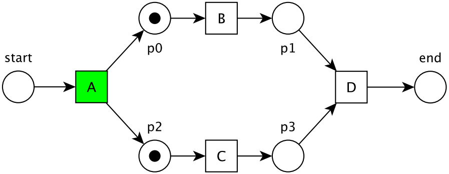
Trace log $(A, 1s)$, $(B, 2s)$, $(C, 4s)$, $(D, 8s)$



Measures

Log-replay examples

Trace log $\color{red}{(A, 1s)}, (B, 2s), (C, 4s), (D, 8s)$

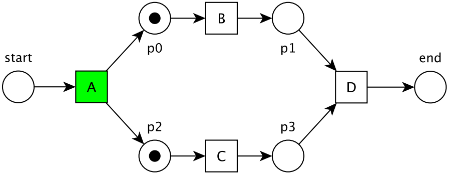


Measures

	p0	p2
$\rtimes$	0	0

Log-replay examples

Trace log $(A, 1s), (B, 2s), (C, 4s), (D, 8s)$

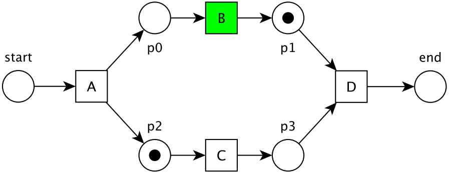


Measures

	p0	p2
$\ltimes$	0	0

Log-replay examples

Trace log $(A, 1s), {\color{red}(B, 2s)}, (C, 4s), (D, 8s)$

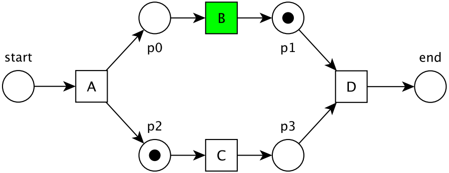


Measures

	p0	p2
$\ltimes$	0	0
$\bowtie$	1	

Log-replay examples

Trace log $(A, 1s), (B, 2s), (C, 4s), (D, 8s)$

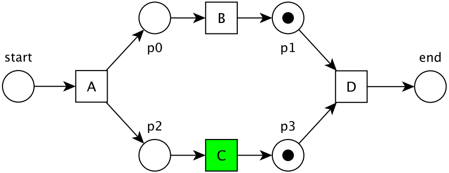


Measures

	p0	p2
$\ltimes$	0	0
$\bowtie$	1	

Log-replay examples

Trace log $(A, 1s), (B, 2s), (C, 4s), (D, 8s)$

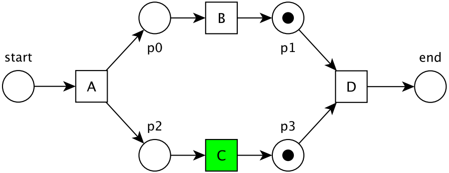


Measures

	p0	p2	p1	p3
$\ltimes$	0	0	2	0
$\bowtie$	1	3		

Log-replay examples

Trace log $(A, 1s), (B, 2s), (C, 4s), (D, 8s)$

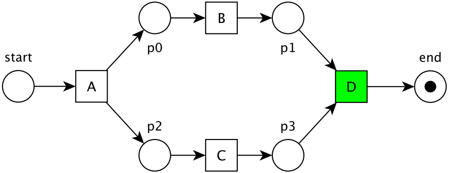


Measures

	p0	p2	p1	p3
$\ltimes$	0	0	2	0
$\bowtie$	1	3		

Log-replay examples

Trace log $(A, 1s), (B, 2s), (C, 4s), (D, 8s)$

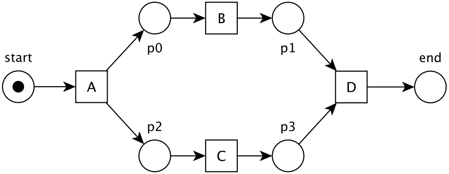


Measures

	p0	p2	p1	p3
$\ltimes$	0	0	2	0
$\bowtie$	1	3	6	4

Log-replay examples

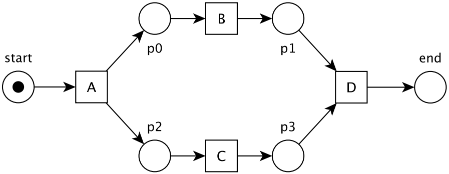
Trace log $(A, 1s), (B, 2s), (D, 8s)$



Measures

Log-replay examples

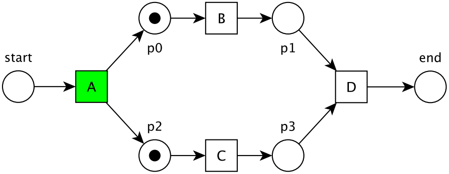
Trace log $(A, 1s), (B, 2s), (D, 8s)$



Measures

Log-replay examples

Trace log $(A, 1s), (B, 2s), (D, 8s)$

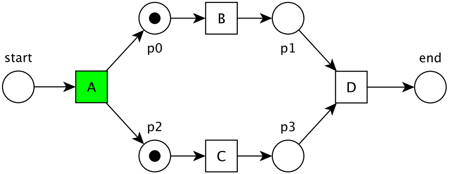


Measures

	p0	p2
$\ltimes$	0	0

Log-replay examples

Trace log $(A, 1s), (B, 2s), (D, 8s)$

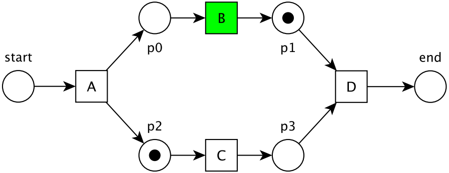


Measures

	p0	p2
⋉	0	0

Log-replay examples

Trace log $(A, 1s), (B, 2s), (D, 8s)$

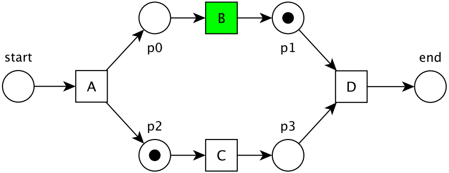


Measures

	p0	p2
$\ltimes$	0	0
$\bowtie$	1	

Log-replay examples

Trace log $(A, 1s), (B, 2s), (D, 8s)$

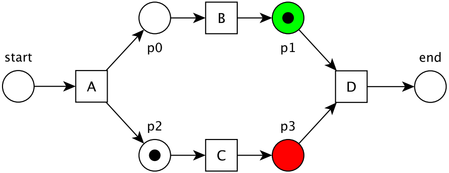


Measures

	p0	p2
$\ltimes$	0	0
$\bowtie$	1	

Log-replay examples

Trace log $(A, 1s), (B, 2s), (D, 8s)$

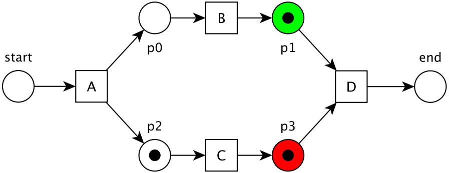


Measures

	p0	p2
$\ltimes$	0	0
$\bowtie$	1	

Log-replay examples

Trace log $(A, 1s), (B, 2s), (D, 8s)$



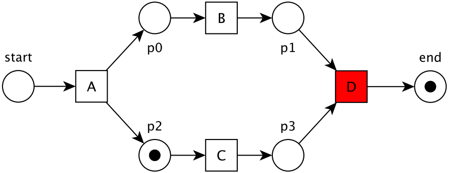
Measures

	p0	p2
$\ltimes$	0	0
$\bowtie$	1	

	p3
Missing	1

Log-replay examples

Trace log $(A, 1s), (B, 2s), (D, 8s)$



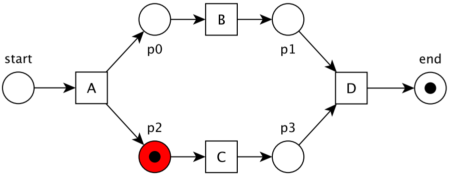
Measures

	p0	p2
$\ltimes$	0	0
$\bowtie$	1	

	p3
Missing	1

Log-replay examples

Trace log $(A,1s),(B,2s),(D,8s)$

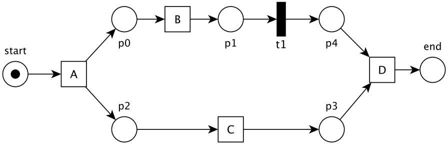


Measures

	p0	p2		p3	p2
$\ltimes$	0	0	*Missing*	1	0
$\bowtie$	1		*Remaining*	0	1

Log-replay examples

Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$



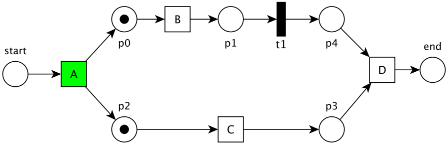
Measures

$\ltimes$

$\bowtie$

Log-replay examples

Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$

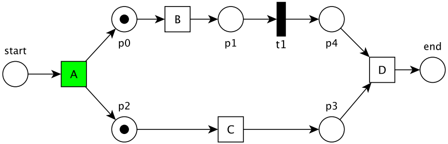


Measures

	p0	p2
$\ltimes$	0	0
$\bowtie$		

Log-replay examples

Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$

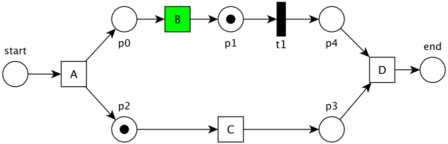


Measures

	p0	p2
⋉	0	0
⋈		

Log-replay examples

Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$

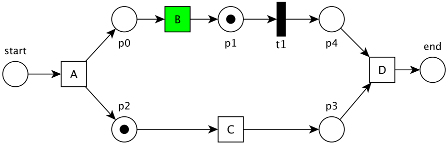


Measures

	p0	p2	p1
$\ltimes$	0	0	0
$\bowtie$	1		

Log-replay examples

Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$

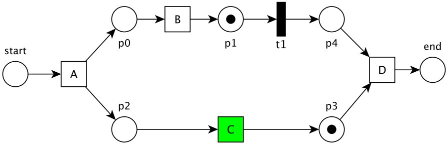


Measures

	p0	p2	p1
$\ltimes$	0	0	0
$\bowtie$	1		

Log-replay examples

Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$

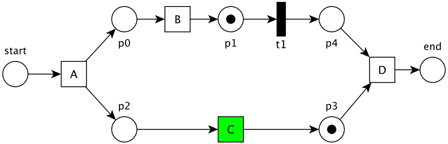


Measures

	p0	p2	p1	p3
$\ltimes$	0	0	0	
$\bowtie$	1	3		

Log-replay examples

Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$

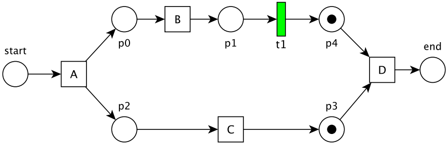


Measures

	p0	p2	p1	p3
$\ltimes$	0	0	0	
$\bowtie$	1	3		

Log-replay examples

Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$

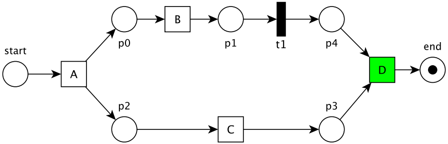


Measures

	p0	p2	p1	p3	p4
$\ltimes$	0	0	0	4	0
$\bowtie$	1	3	6		

Log-replay examples

Trace log $(A, 1s), (B, 2s), (C, 4), (D, 8s)$



Measures

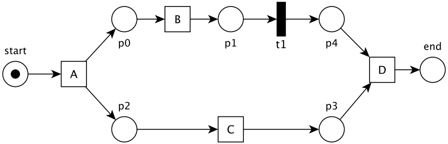
	p0	p2	p1	p3	p4
$\ltimes$	0	0	0	4	0
$\bowtie$	1	3	6	4	0

A refined Performance analysis

- Exploit standard log-replay techniques in order to reuse existing sw infrastructure
- Transform resulting transition list into eager sequences $R = [tr_1, ..., tr_n]$ for each tr_i invisible transition
  - let tr_p the last preceding ($p < i$) visible transition
  - $\bullet tr_i \cap tr_p \bullet \neq \emptyset$
- Straightforward algorithm: for each invisible transition tr_i
  1. left shift the transition until a visible transition $\bullet tr_i \cap tr_p \bullet \neq \emptyset$ is found
- All conformance metrics are not changed

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 $A, B, t1, C, D$



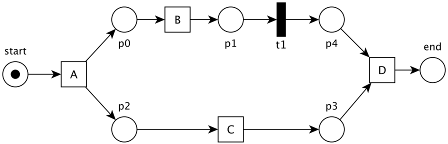
Measures

$\ltimes$

$\bowtie$

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 $A, B, t1, C, D$



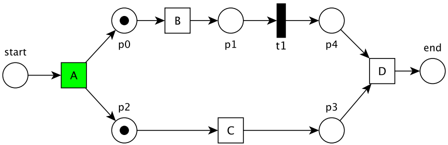
Measures

$\ltimes$

$\bowtie$

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 $A, B, t1, C, D$

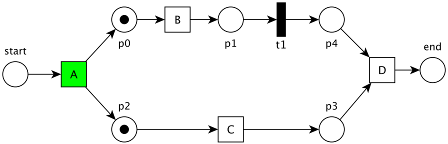


Measures

	p0	p2
$\ltimes$	0	0
$\bowtie$		

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 $A, B, t1, C, D$

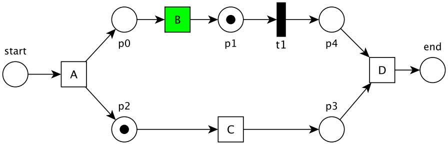


Measures

	p0	p2
$\ltimes$	0	0
$\bowtie$		

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 $A, B, t1, C, D$

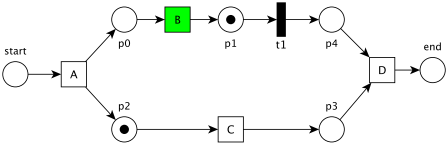


Measures

	p0	p2	p1
$\ltimes$	0	0	0
$\bowtie$	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 $A, B, t1, C, D$

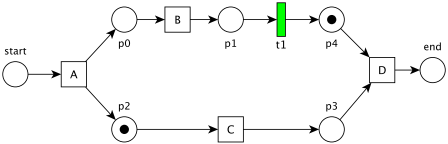


Measures

	p0	p2	p1
$\ltimes$	0	0	0
$\bowtie$	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 $A, B, t1, C, D$

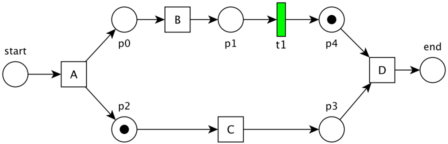


Measures

	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 $A, B, t1, C, D$

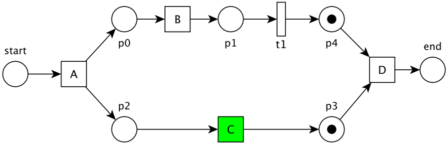


Measures

	p0	p2	p1	p3	p4
$\ltimes$	0	0	0		
$\bowtie$	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 $A, B, t1, C, D$

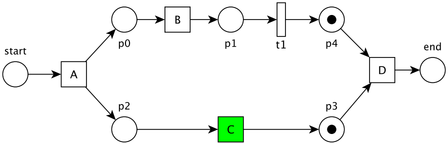


Measures

	p0	p2	p1	p3	p4
$\ltimes$	0	0	0	0	2
$\bowtie$	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 $A, B, t1, C, D$

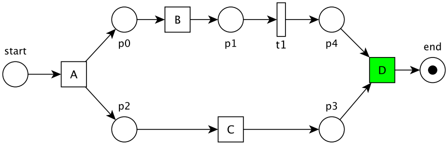


Measures

	p0	p2	p1	p3	p4
$\ltimes$	0	0	0	0	2
$\bowtie$	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 $A, B, t1, C, D$

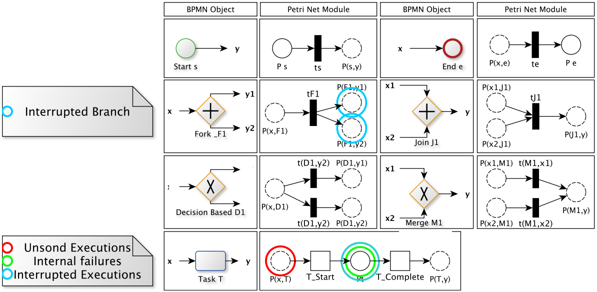


Measures

	p0	p2	p1	p3	p4
$\ltimes$	0	0	0	0	2
$\bowtie$	1	2	0	4	6

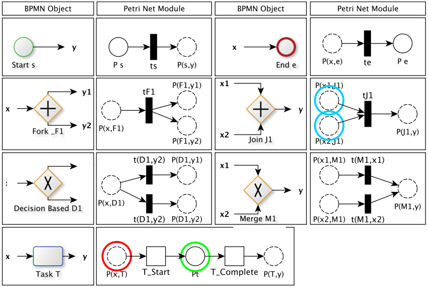
From Analysis to BPMN (Conformance)

- Missing tokens: Log replay produces missing tokens only to fire visible transitions ⇒ pre-set of at least one visible transition
- Remaining tokens fires invisible transitions fired only if required by visible transition ⇒ places in the post-set of a visible transition or of an invisible transition spawning several tokens



From Analysis to BPMN (Performance)

- Waiting time: invisible transitions fired immediately ⇒ pre-set of visible transitions
- Synchronization time places having at least one transition in their post-set that depends on another place.



○ Synchronization time

○ Activation Time
○ Execution Time

Theoretical results

- Refined PetriNet techniques to handle invisible transitions
- Measurement projection on the BPMN model

Development results

- A new ProM context to execute the plug-ins into a GUI-less environment
- New ProM plug-ins
  - Model transformation from BPMN to PetriNet
  - Eager Sequence transformer
  - PetriNet performance evaluator
  - Projection of PetriNet measures to the starting BPMN model

1. Introduzione
   - Business Process Management
   - Sommario
2. Supporto alla modellazione tramite BPMN
   - From BPMN to Petri Net
3. The The Process Monitoring platform
   - Engineering efforts
4. Formal Analysis based on Petri nets
   - Adopted Petri Net techniques
   - A refined Performance analysis
   - From Analysis to BPMN
5. Concluding Remarks