

University of Pisa

Business Process Modeling

E-learning Project

2022/2023

Team:

Maria Grazia Antico 645005
Graziano Amodio 562357

Professor:

Roberto Bruni



UNIVERSITÀ DI PISA

Indice

1	Introduzione	1
2	BPMN Model	1
2.1	Calendario	2
2.2	Lezione	3
2.3	Chat	3
2.4	Ultima Lezione	3
2.5	Variante	4
3	Net analysis	4
3.1	Studente	5
3.2	Didattica	7
4	Workflow System	10
4.1	Originale	10
4.2	Variante	10

1. Introduzione

Il problema presentato illustra lo scenario di un corso di apprendimento personalizzato offerto in teledidattica, descrivendo il processo che coinvolge lo studente che interagisce con la didattica per poter usufruire dei corsi messi a disposizione. In particolare, sono stati inizialmente individuati gli attori coinvolti, rappresentati dai seguenti:

- **Studente**
- **Segreteria**
- **Referente**

Il processo è stato modellato utilizzando il linguaggio BPMN in quanto si basa su una *Choreography* che interagisce con due diversi business process, sfruttando **BPMN.io** per il disegno del diagramma. Per quest'ultimo sono stati creati *due pool* (**studente e didattica**) che interagiscono tra loro scambiandosi informazioni attraverso i *message flow arrows*. Il pool didattica è composto, inoltre, da due *lanes* che permettono di contraddistinguere i task della **Segreteria** e quelli del **Referente**.

2. BPMN Model

In generale, il processo parte dall'attività dello studente il quale, richiedendo informazioni sui corsi, avvia l'istanza della segreteria, la quale dopo aver interagito con lo studente con una serie di attività (relativi alla lista di corsi disponibili e scelta di uno di essi), seleziona un referente da assegnare andando ad attivare il processo di quest'ultimo. Lo studente paga dunque la prima rata e si mette in contatto con il referente il quale propone un calendario. Tra i due vi sono varie interazioni relative all'accordo del calendario definitivo, rappresentate dal sotto processo "**Calendario**". Una volta definito il calendario, viene fissato l'appuntamento, descritto dal sotto processo "**Lezione**", in cui il referente espone i concetti e risponde ad eventuali domande dello studente, ed a fine domande viene assegnato un esercizio da svolgere allo studente. Dopo l'incontro il referente carica materiale ed esercizi nella piattaforma dedicata, e prima dell'inizio di un altro incontro vi è nuovamente uno scambio di interazioni tra i due attraverso una chat apposita, mostrati nel sotto-processo "**Chat**". Il referente, inoltre, comunica se la lezione successiva sarà l'ultima o meno e, in quest'ultimo caso, si ha il sotto processo "**Ultima lezione**" in cui il referente pone degli esercizi allo studente fino a quando non arriva alla formulazione di un giudizio, comunicandogli l'esito finale che verrà trasmesso in segreteria e registrato, permettendo la consegna dell'attestato allo studente dopo che quest'ultimo avrà saldato la seconda rata.

Il diagramma finale ottenuto è il seguente, mostrato in Figura 1:

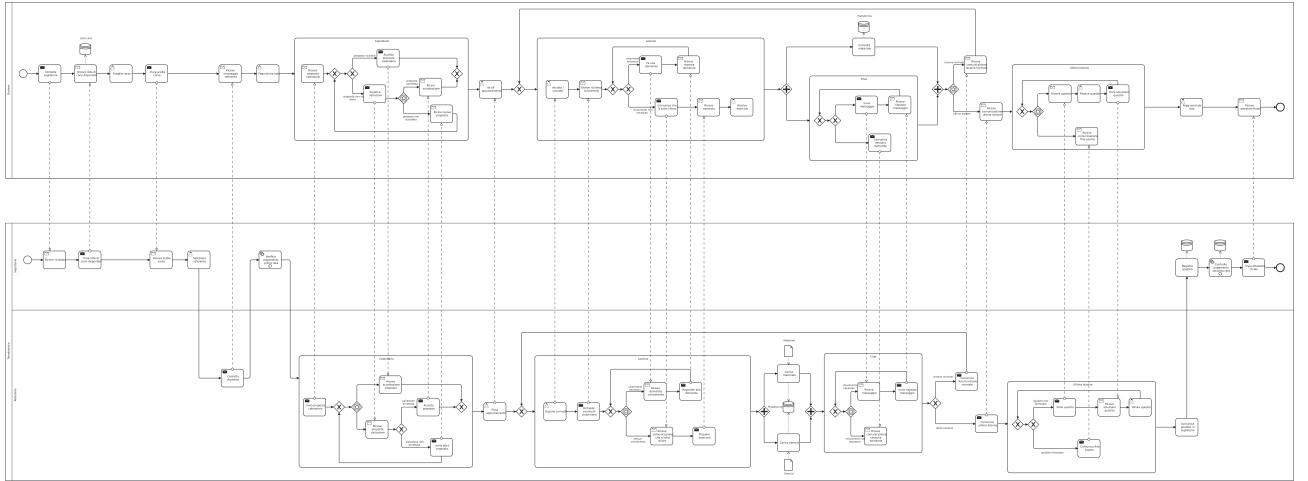


Figura 1: BPMN diagram

Si sono utilizzati alcuni **Artefacts** come, per esempio, i *Databases* per rappresentare la lista corsi, i pagamenti e la piattaforma didattica (dove vengono inseriti i file di esercizi e materiale da parte del referente). In seguito saranno descritti più dettagliatamente i singoli sotto-processi, sia per quanto riguarda lo studente che il referente seppur siano simili tra loro.

2.1 Calendario

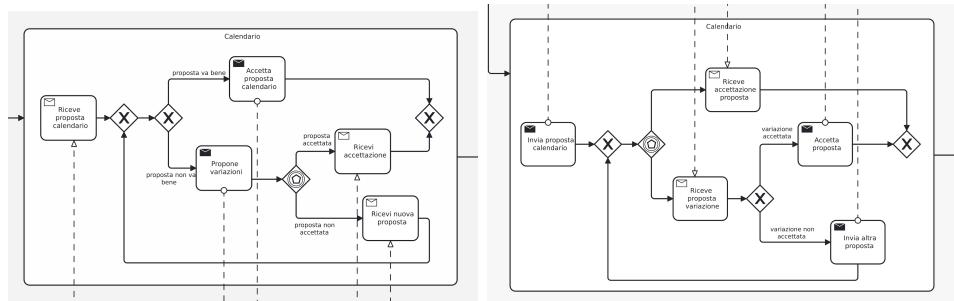


Figura 2: Sotto-processo "calendario" Studente (sx) e Referente (dx)

Dal punto di vista dello studente, quest'ultimo riceve una proposta di calendario che può o accettare subito, terminando dunque il sotto-processo, o proporre variazioni al referente mettendosi in attesa (utilizzando un event-based gateway) di risposta negativa o positiva fino a quando non si arriva ad un calendario adatto ad entrambi. Dal punto di vista del referente, invece, dopo aver inviato una proposta, si mette in attesa anch'egli di una risposta e, nel caso di una proposta di variazioni, sceglierà se accettare la proposta o, in caso contrario, inviare una nuova proposta allo studente fino a quando non si arriva ad un accordo comune.

2.2 Lezione

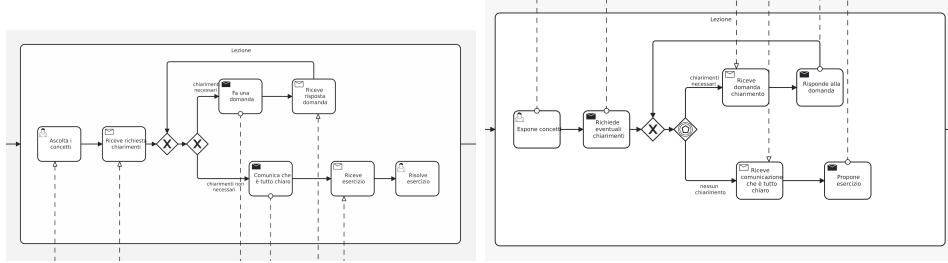


Figura 3: Sotto-processo "Lezione" Studente (sx) e Referente (dx)

Dal punto di vista dello studente, dopo aver ascoltato i concetti può comunicare al referente di non aver domande, terminando il sotto-processo dopo aver ricevuto un esercizio ed averlo risolto, oppure in caso di dubbi può fare una domanda, fino a quando non è tutto chiaro. Dal punto di vista del referente, dopo aver esposto dei concetti, chiede allo studente se necessiti di alcuni chiarimenti, mettendosi in attesa di risposta fino a quando lo studente non comunichi che è tutto chiaro, assegnandogli infine un esercizio e chiudendo il sotto-processo.

2.3 Chat

Dal punto di vista dello studente, tra una lezione e l'altra può scegliere se inviare un messaggio al referente tramite chat apposita o, nel caso di nessuna eventuale domanda, il sotto-processo è concluso. Per quanto riguarda il referente, si mette in attesa di risposta fino a quando lo studente non ha nessuna domanda.

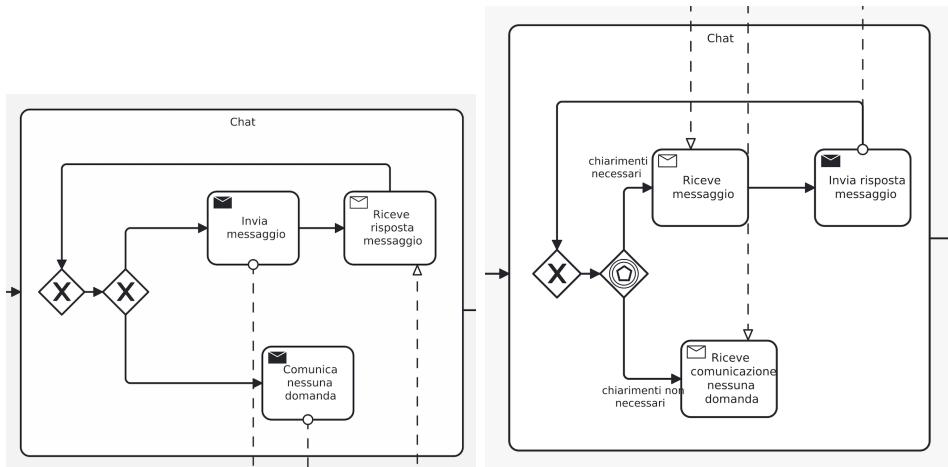


Figura 4: Sotto-processo "Chat" Studente (sx) e Referente (dx)

2.4 Ultima Lezione

Per quanto riguarda lo studente, dopo aver ricevuto la comunicazione del referente di ultima lezione, si mette in attesa di risposta in base al fatto se il referente abbia un giudizio già formulato o meno. In caso di esito non formulato, riceve un quesito e lo

risolve fino a quando non riceve una comunicazione di fine esame, terminando il sotto-processo. Dal punto di vista del referente, dopo aver comunicato che ci sarà l'ultima lezione, in caso di giudizio già formulato il sotto-processo termina con la comunicazione di fine esame, altrimenti viene posto un quesito fino a quando non viene raggiunto un giudizio.

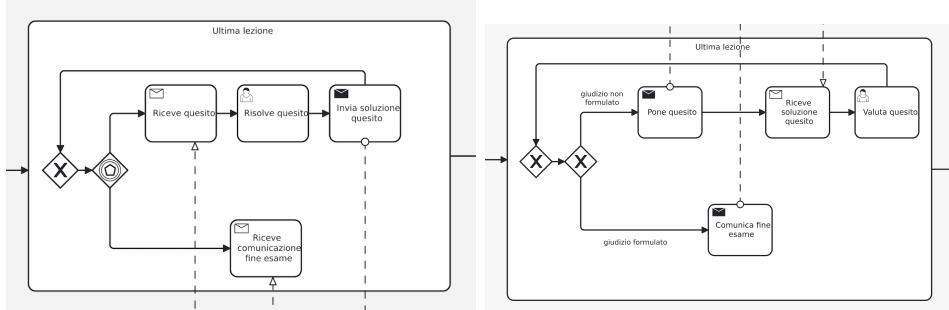


Figura 5: Sotto-processo "Ultima lezione" Studente (sx) e Referente (dx)

2.5 Variante

La versione Variante consiste nel fare in modo che al termine del corso lo studente possa scegliere se iniziare un nuovo corso. È stato dunque inserito uno XOR-gateway nel pool Studente con cui quest'ultimo può comunicare se vuole seguire un altro corso o meno, portando eventualmente alla ripetizione dell'intero processo a partire dalla ricezione della lista dei corsi. Similmente, nel pool Didattica nel lane della Segreteria, si è aggiunto un event-based gateway per ricevere la comunicazione della scelta dello studente a seguire o meno un nuovo corso.

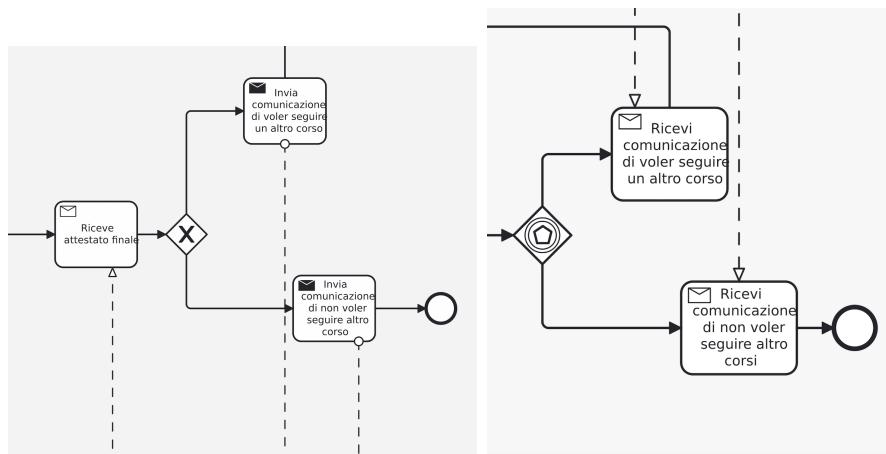


Figura 6: Sotto-processo "Ultima lezione" Studente (sx) e Referente (dx)

3. Net analysis

Il diagramma BPMN è stato trasformato in seguito in un Petri net utilizzando la versione "desugarized". Si sono inizialmente trasformati i singoli pool in Workflow Net

e, successivamente, sono stati uniti i module formando il Workflow System finale. Per ognuno di essi si è svolta l'analisi semantica e ricavato anche il Coverability graph. Il tutto è stato svolto su WoPed, con il supporto di Woflan per quanto riguarda l'analisi dei Workflow Systems e per la verifica preliminare della condizione di soundness, prima di procedere all'analisi semantica su WoPed che ha richiesto molte ore di attesa. Nelle sezioni successive verrà approfondita l'analisi semantica di ognuno di essi.

3.1 Studente

Si riportano di seguito, sia per la versione originale che per la variante dello Studente, l'analisi semantica, la petri net e il coverability graph.

Come si può notare dalle immagini dei *coverability graph* (Figura 8), e confermato dall'analisi semantica (Figura 7), i due workflow rispettano tutte e tre le condizioni necessarie per essere **sound**: non ci sono *dead task*, in ogni punto del grafico c'è sempre un marking M primo che porta alla fine e non vi è la possibilità che un token rimanga lungo il percorso una volta raggiunto il nodo finale. La struttura della Net originale è la seguente: **49 places**, **53 transitions** e **108 archi**. Per la versione della variante, si hanno invece **54 places**, **59 transitions** e **122 archi**.

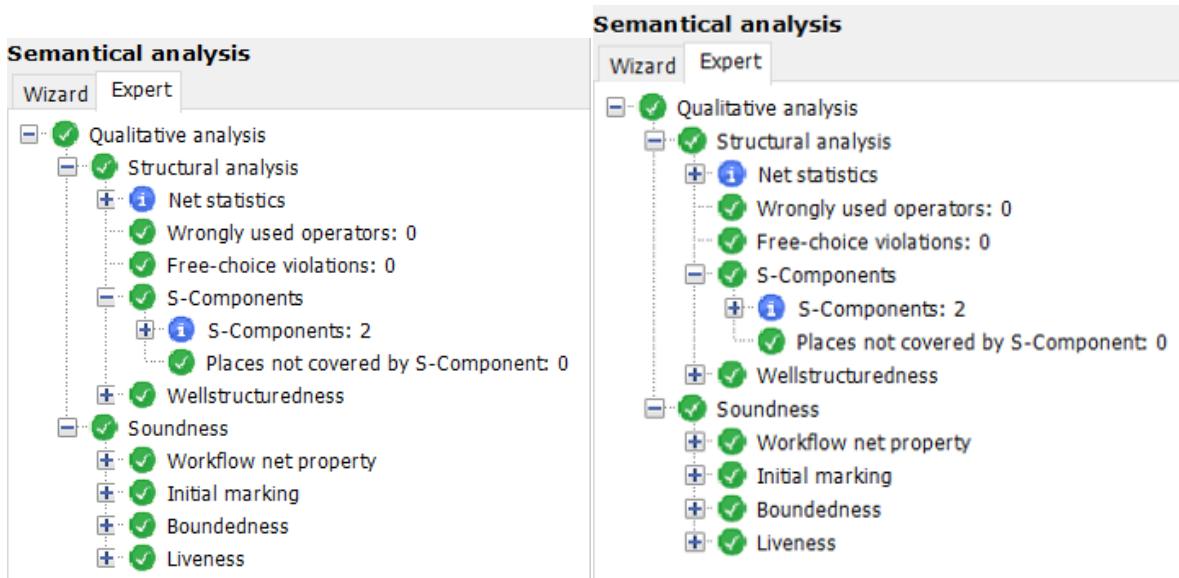


Figura 7: Semantical Analysis - student original (sx) e student variant (dx)

I *Coverability graph* hanno rispettivamente **54 vertici** e **66 edges** per la versione originale, per la variante **59 vertici** e **72 edges**.

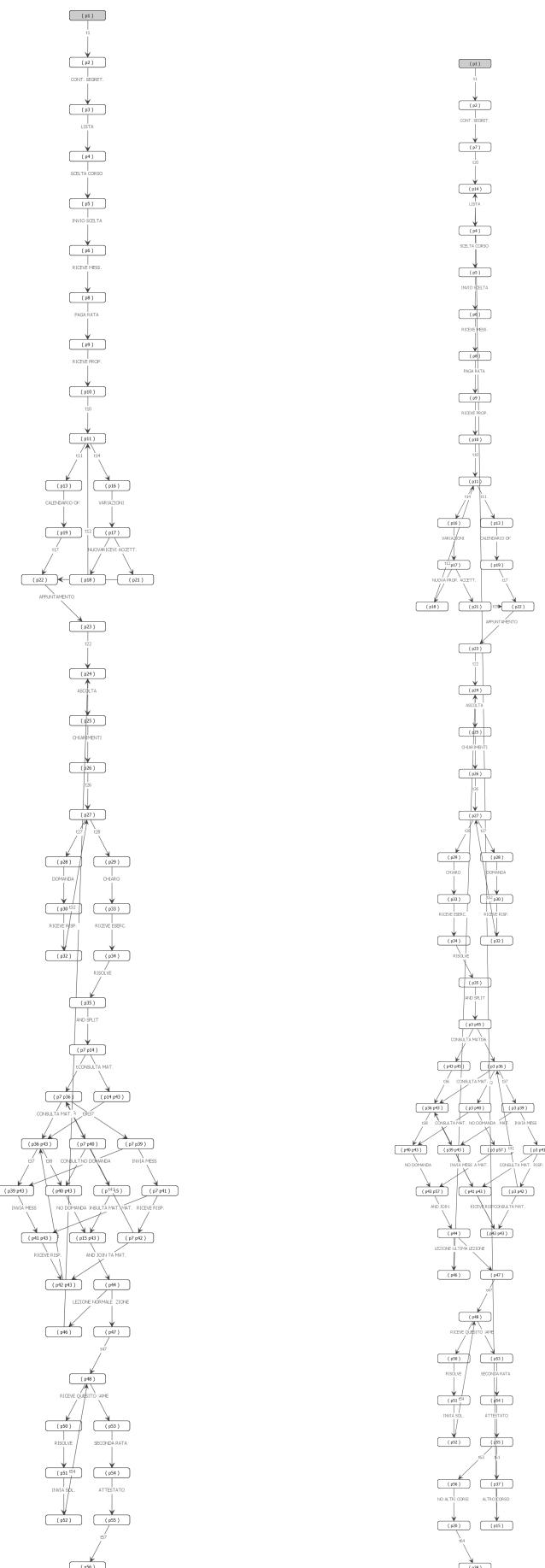


Figura 8: Coverability graph - student original (sx) e student variant (dx)

Per entrambe le Net (originale e variante) si verificano le stesse condizioni. Sono **Workflow Net** in quanto composte da un place iniziale, un place finale e tutti i places e transistions appartengono ad un percorso tra il place iniziale e finale. Sono **free-choice** ma non sono **né S-system e né T-system**. Non sono *S-system* solo a causa dell'AND-split seguito dall'AND-join, e dunque è possibile trovare facilmente una positive *S-invariant* per entrambe. Di fatto, considerando $I(p)=2$ per ogni place fuori dall'AND-split ed AND-join e per quelli intorno ad essi $I(p)=1$ otteniamo una *S-invariant*. Non essendo *S-net* sono dunque anche **sound**. In entrambi si notano **2 S-Component** che insieme coprono tutti i places e transitions su entrambe le Net, confermando la nozione di **S-coverable**. Non sono **neanche T-Net** a causa della presenza degli event-based gateway, tradotti da places aventi più di un arco in uscita. Le Net sono inoltre **well-structured**, in quanto **non ci sono PT-handles o TP-handles**. Si può anche affermare che è **safe**, osservabile anche dai coverability graph che presentato solo nodi con places con $M(p)=1$.

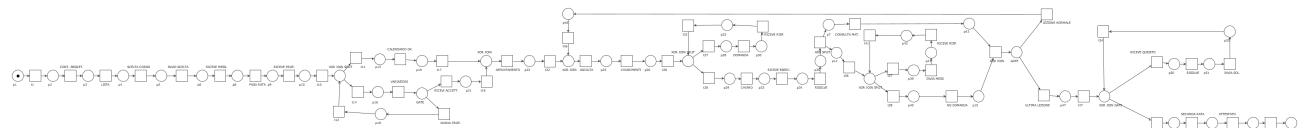


Figura 9: Net student originale

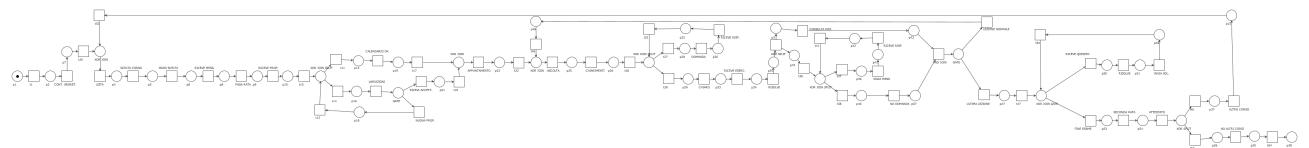


Figura 10: Net student variante

3.2 Didattica

L'analisi di Didattica è identica a quella di Studente, unica differenza è il numero di *places*, *transitions* ed elementi del *coverability graph*. Di fatto, il *coverability graph* per la versione originale possiede **51 vertici** e **57 edges** mentre per la variante si hanno **54 vertici** e **61 edges**.

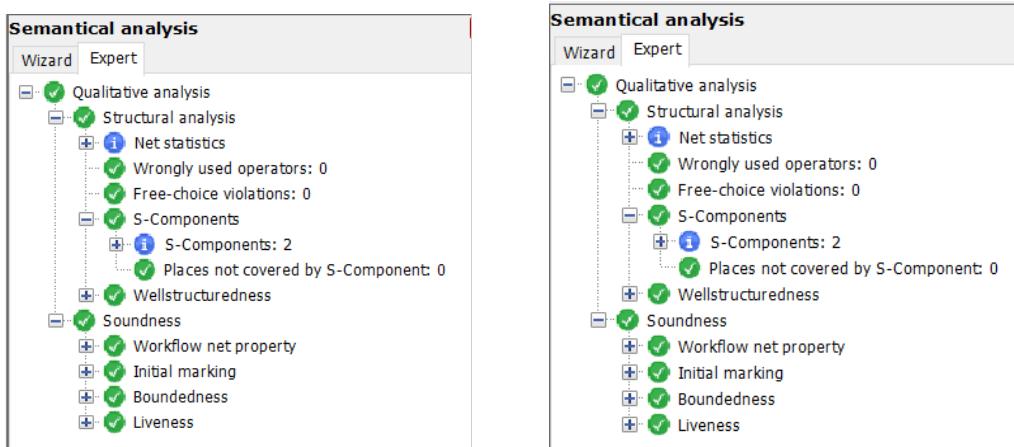


Figura 11: Semantical Analysis - didattica original (sx) e didattica variant (dx)

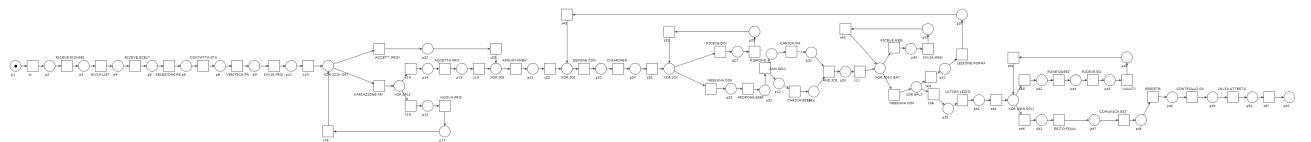


Figura 12: Net didattica originale

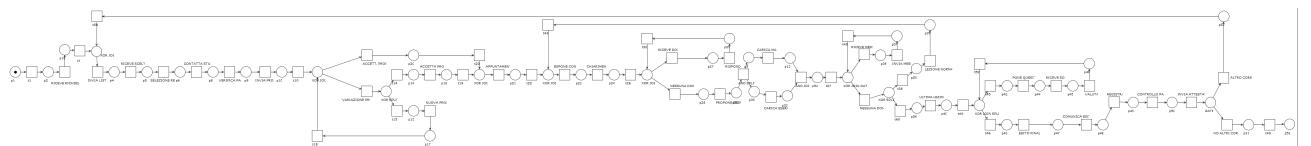


Figura 13: Net didattica variante

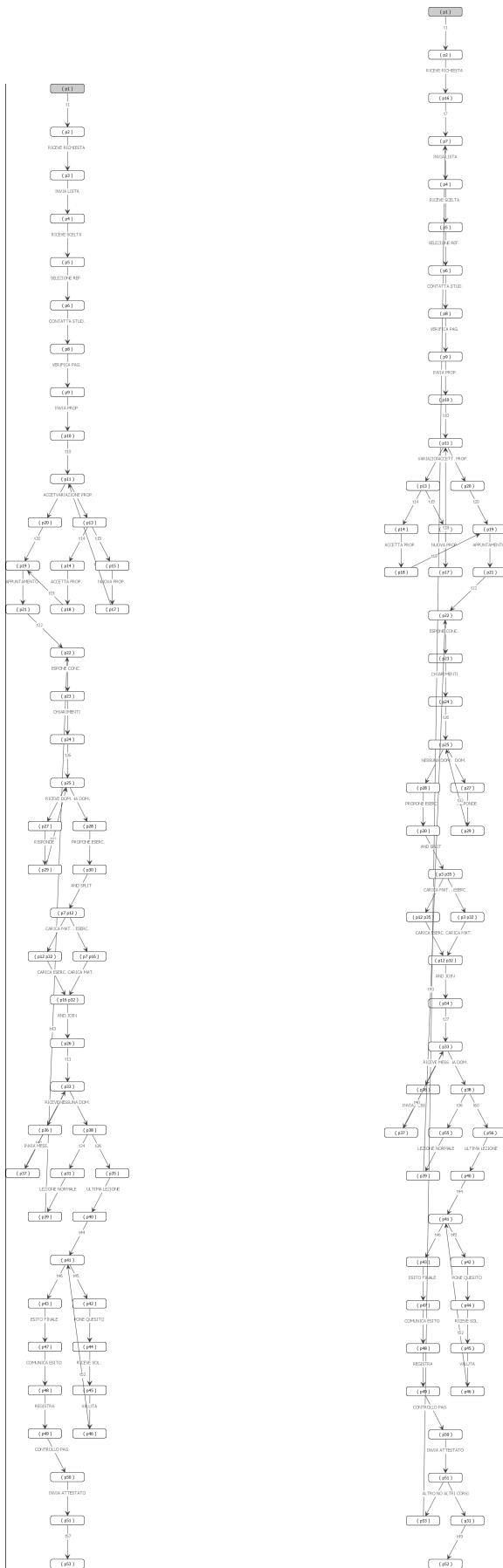


Figura 14: Coverability graph - didattica originale (sx) didattica variante (dx)

4. Workflow System

Per unire le due Net in un singolo modulo è stato impostato un place finale comune, collegato attraverso un AND join ed il place iniziale comune è rappresentato dal punto iniziale dello Studente in quanto è quest'ultimo colui che avvia il processo. Sono stati, infine, tradotti i vari message flows aggiungendo places addizionali per collegare le relative transitions.

4.1 Originale

Inizialmente, l'analisi del Workflow System è stata fatta con Woflan per verificare preventivamente la condizione di soundness. In seguito, si è riusciti ad avere l'analisi semantica con WoPed come visibile di seguito:

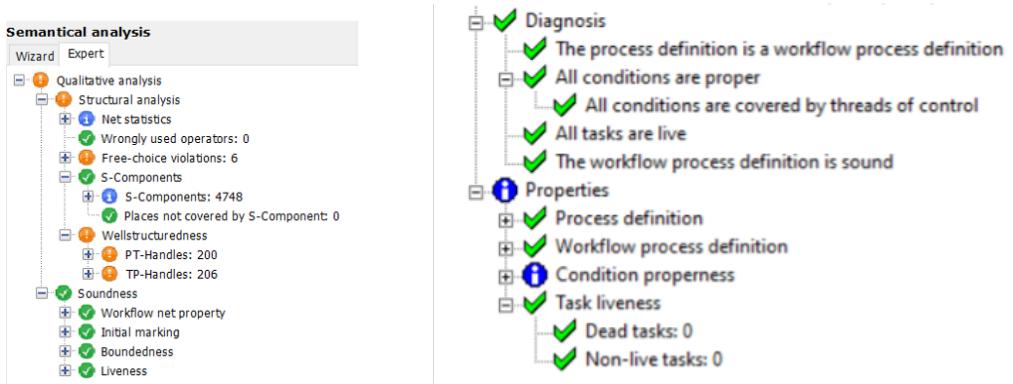


Figura 15: Semantical Analysis WF system original WoPed (sx) system original Woflan (dx)

Come è possibile notare, la condizione di **soundness** è **mantenuta** anche nel Workflow System, **ma non è più free-choice** a causa degli **event-based gateway** che, in questo caso, introducono coppie di transitions i cui pre-sets non sono strettamente uguali e nemmeno disgiunti. Si hanno, di fatto, 6 violazioni free-choice che corrispondono anche al numero di event-based gateway presenti. È anche possibile osservare che la Net **non è più well-structured** come nelle precedenti singole Net, in quanto sono presenti **numerosi PT-handles e TP-handles**. Il numero delle **S-components**, inoltre, è cresciuto in maniera esponenziale a 4748. Nonostante ciò, la Net è comunque **S-coverable** quindi esiste almeno una positive S-invariant.

4.2 Variante

In seguito si riportano, inoltre, i risultati della *versione variante* ottenuti da Woflan da cui risulta anche qui che: la Workflow Net è definita correttamente (soddisfando le tre proprietà); **non esistono unbounded sequences** e dunque si può affermare che N^*

è bounded (verificato attraverso i 7768 thread di controllo); non esistono non-live sequences e dunque possiamo affermare che N^* è live; è sound essendo N^* live e bounded per il Main Theorem; non è free-choice.

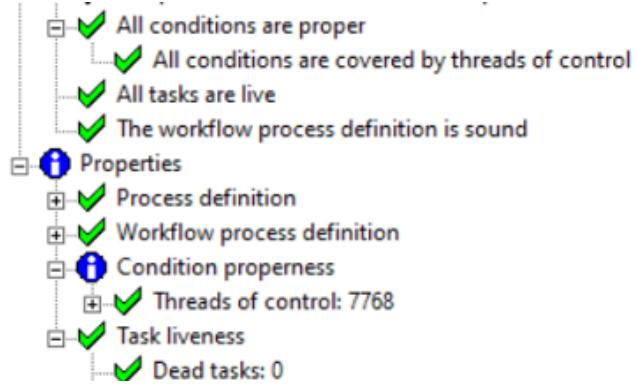


Figura 16: Analysis Woflan - Variant version

Si riportano, di seguito, la Petri Net completa originale e la versione variante.

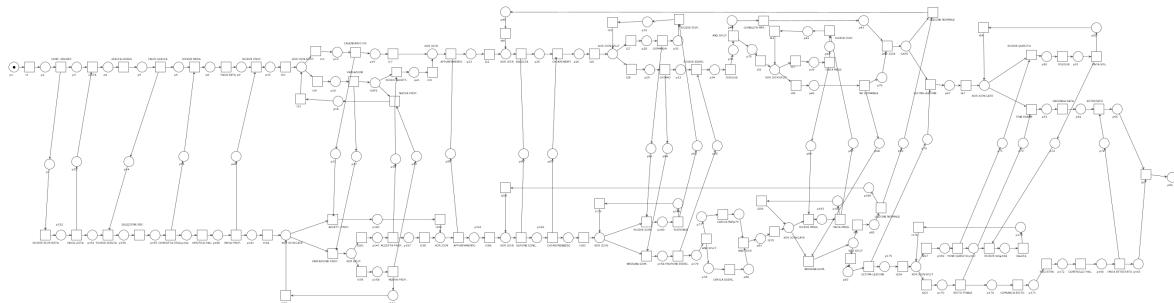


Figura 17: Petri Net Completa Originale

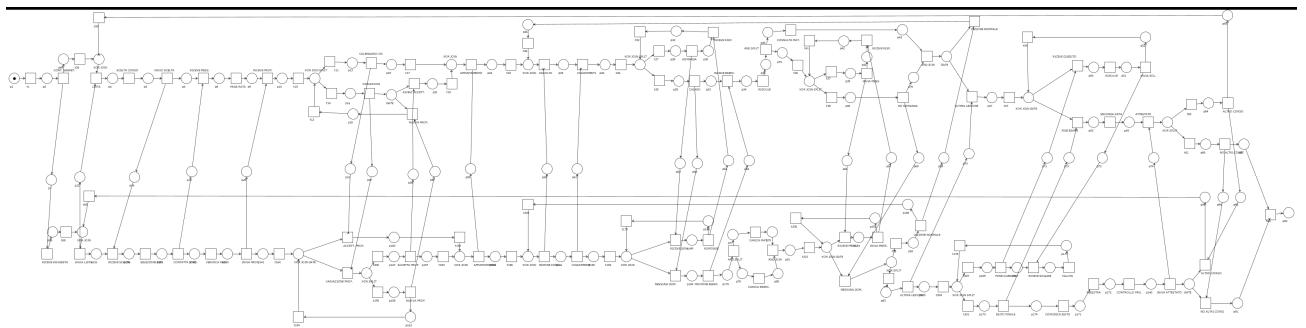


Figura 18: Petri Net Completa Variante

Infine, si riportano i Coverability graphs della versione originale e della variante del Workflow system.

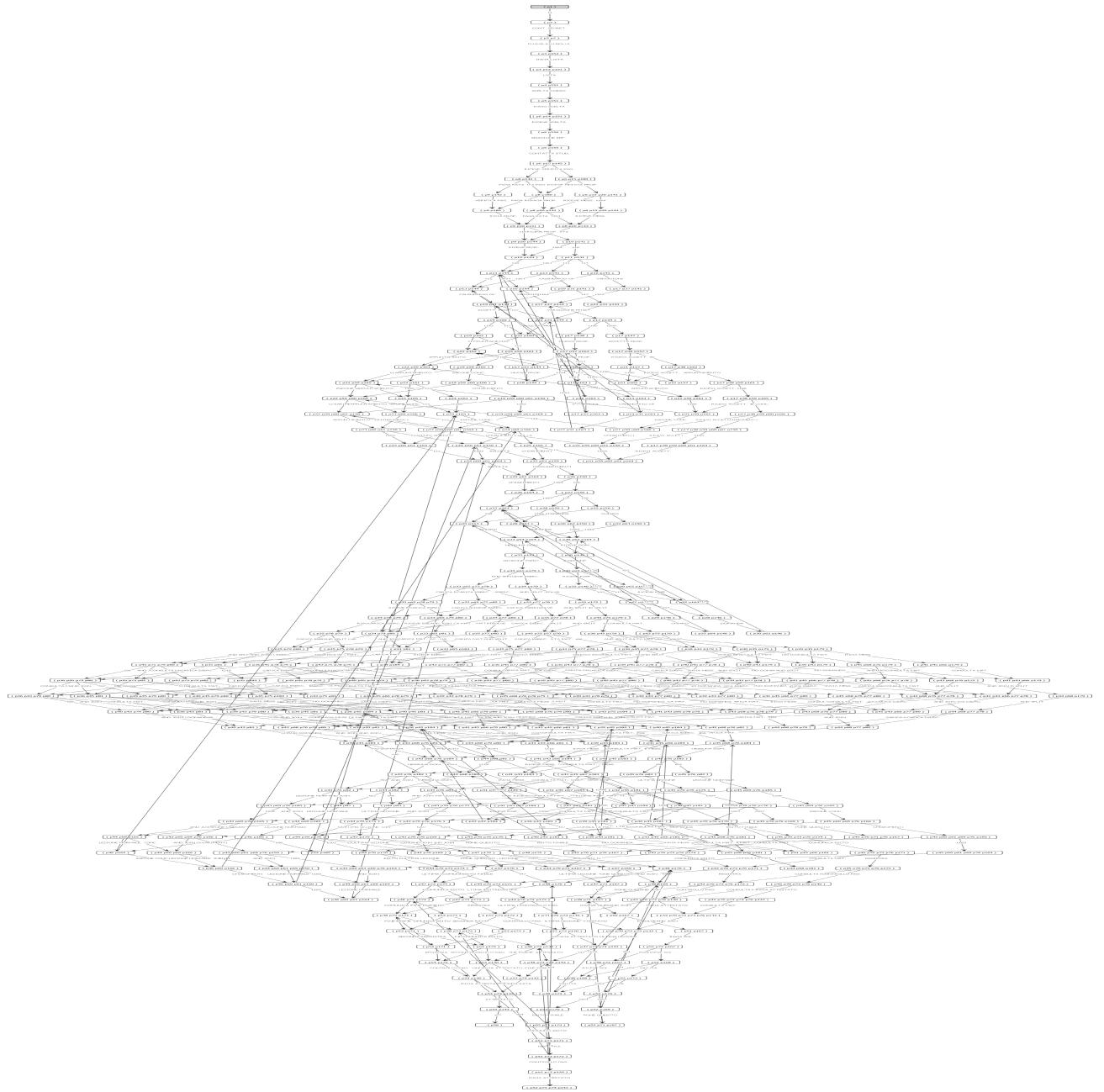


Figura 19: Coverability graph WF - originale

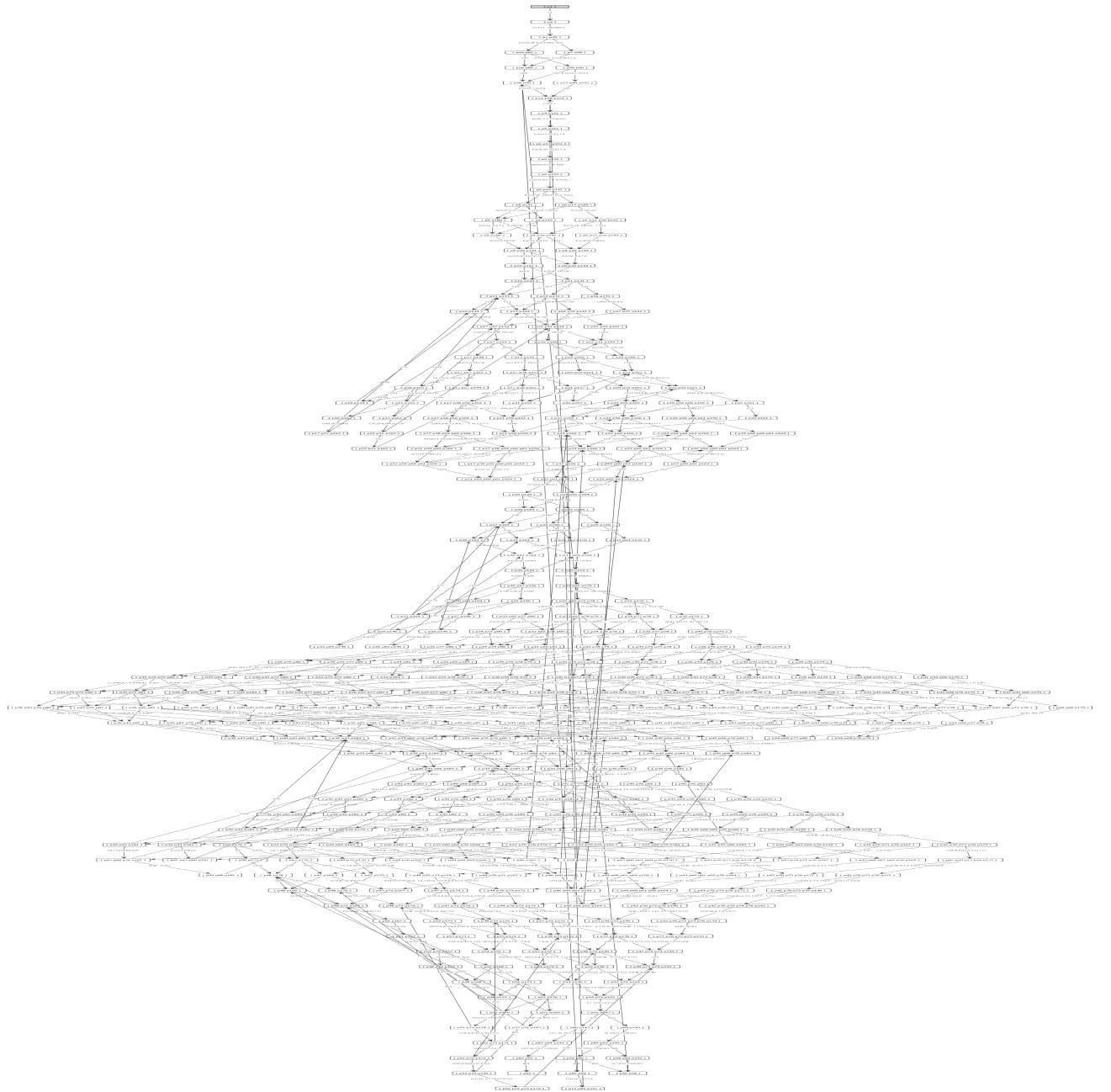


Figura 20: Coverability graph WF - variant