

UNIVERSITÀ DI PISA



DIPARTIMENTO DI INFORMATICA
Laurea Magistrale in Data Science and
Business Informatics

Business Process Modeling

Turni di ferie

2021/2022

Autori

Giulia Marcoccio
Giovanni Scognamiglio

Contents

1	Introduzione	1
2	Analisi agenti	2
2.1	Impiegato 1	2
2.2	Impiegato 2	3
2.3	Sistema elettronico	4
2.4	Reachability graphs	6
3	Collaborazione tra gli agenti	8
3.1	BPMN	8
3.2	Petri Net	8
4	Caso Alternativo	12
4.1	BPMN	12
4.2	Petri Net	12

1 Introduzione

L'analisi presentata in questo report concerne la progettazione dello scenario nel quale due impiegati devono stabilire il loro piano ferie per l'anno venturo. Gli impiegati possono scegliere una data e decidere se selezionarla o deselectionarla dalla memoria temporanea del piano, oppure azzerare la loro scelta e ritornare all'inizio della compilazione. Questo processo viene gestito da un sistema elettronico che deve controllare i giorni selezionati dagli impiegati e verificare che ci sia la disponibilità di almeno una persona per ogni giorno lavorativo.

Nel caso in cui si presentassero dei conflitti, ossia la sovrapposizione di uno o più giorni di ferie nei piani, il sistema è in grado di segnalarli avviando una conversazione tra gli impiegati, i quali potranno scambiarsi a turno dei messaggi e trovare un accordo. Il modello prevede che il primo impiegato inizi la conversazione e ciclicamente i due utenti potranno scegliere se continuare la conversazione tramite un sistema di proposta/controproposta, oppure accettare la proposta dell'altro. Dopo aver trovato un accordo potranno nuovamente modificare i loro piani ferie e il processo terminerà solo una volta che i piani ferie verranno entrambi approvati dal sistema elettronico.

Perciò, gli attori coinvolti sono tre, ossia i due impiegati e il sistema elettronico che fa da intermediario tra i due approvando i piani ferie e favorendo la comunicazione tra i due utenti, i quali presenteranno la stessa forma per la sezione riguardante la scelta dei giorni da inserire nel piano ferie. In seguito, invece, in caso di conflitti, sarà il primo impiegato ad avviare la conversazione in quanto non sarebbe stato possibile avviarla simultaneamente per entrambi gli utenti. Per questo motivo il secondo impiegato dovrà attendere la ricezione del messaggio per rispondere e entrambi, a turno, potranno decidere se effettuare una proposta oppure accettare quella del collega finché non si troverà una soluzione e entrambi verranno ricondotti al punto in cui potranno decidere se modificare il loro piano, ossia selezionare o deselectionare uno o più giorni di ferie o azzerare completamente il piano, oppure sottoporlo nuovamente a verifica da parte del sistema elettronico.

Inizialmente, il processo di modellazione è stato realizzato tramite l'utilizzo della notazione BPMN in quanto si presta maggiormente a rappresentare la coreography. Per rappresentare il BPMN è stato creato il diagramma con **Bpmn.io**, mentre le Petri Nets dei singoli attori e del sistema generale sono state generate e analizzate con **WoPeD** e **Woflan**.

2 Analisi agenti

In primo luogo avverrà un'analisi dei singoli attori in modo da poter comprendere i loro comportamenti all'interno del sistema totale. Per ciascun agente verrà presentato il diagramma BPMN e la relativa Petri net, seguita dall'analisi semantica generata da WoPeD.

Si è optato per un'organizzazione in 3 pools, uno per ciascun agente coinvolto nel processo. Per ciascun agente, le fasi principali del processo sono tre:

1. La composizione del piano ferie;
2. L'invio e approvazione del piano ferie;
3. L'eventuale fase di conflitto e avvio della conversazione, oppure di fine del processo.

I due impiegati iniziano contemporaneamente la compilazione del piano e comunicano esclusivamente attraverso il sistema elettronico che funge da intermediario tra i due.

2.1 Impiegato 1

Il primo impiegato, una volta effettuato l'accesso al sistema, può iniziare a creare il suo piano ferie. Questo processo ha due opzioni iniziali gestite da uno Xor-split, l'utente può scegliere di selezionare o deselectare una determinata data nella memoria temporanea del sistema elettronico, oppure azzerare il suo piano nel caso in cui voglia cancellare tutte le opzioni memorizzate. Dopo ogni selezione o deselezione di una data, egli può scegliere tramite un ciclo gestito da uno Xor-split, di reiterare il processo e modificare ulteriormente il suo piano ferie (dove potrà effettuare una nuova scelta oppure azzerare il piano), oppure, nel caso in cui abbia raggiunto il piano ferie desiderato, può sottoporlo ad approvazione.

Tramite un event-based gateway, l'utente rimarrà in attesa della valutazione del proprio piano ferie da parte del sistema. Nel caso in cui il piano ferie venga approvato, l'utente riceverà la notifica di approvazione e potrà concludere il proprio processo. In caso contrario, l'utente riceverà dal sistema le date in cui si presenta un conflitto con l'altro impiegato. L'impiegato 1 invierà quindi un messaggio con una prima proposta all'impiegato 2 e tramite un event-based gateway, rimarrà in attesa della decisione dell'impiegato 2.

In caso quest'ultimo accetti la proposta, l'impiegato 1 riceverà una notifica di accettazione della proposta e verrà rimandato alla schermata dove potrà modificare il suo piano ferie se necessario e far approvare nuovamente il piano dalla macchina.

Nel caso in cui l'impiegato 2 non accetti la proposta e risponda con un messaggio di controproposta, l'impiegato 1 riceverà il messaggio di controproposta e potrà a sua volta accettare la controproposta (e ritornare alla schermata dove può modificare il piano se necessario e inoltrarlo nuovamente alla macchina) oppure rispondere con un nuovo messaggio di proposta e tornare al loop di messaggistica.

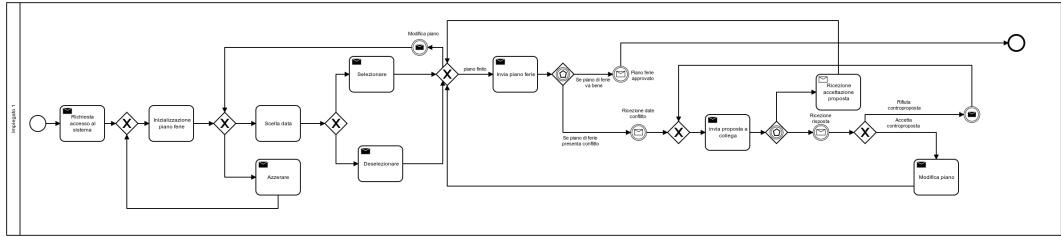


Fig. 1: BPMN Impiegato 1

In seguito, è stata effettuata la traduzione in Petri net in alcuni step: la creazione di places al posto di archi e di transizioni che rimpiazzano attività e eventi; traduzione dei gateways e infine creazione del place iniziale e di quello finale. Sono stati così ottenuti 10 places, 15 transizioni e 30 archi. Si tratta di un Workflow net in quanto presenta un place iniziale i ed uno finale o e ogni place e transizione appartiene ad un percorso tra i e o . La rete è S-net e rispetta tutte le proprietà della soundness, infatti non esistono dead task (come si può verificare nel coverability graph), e sono soddisfatte le proprietà di option to complete e proper completion. Come si può vedere dalla Figura 3 la rete è deadlock-free, bounded e free-choice e wellstructuredness, infatti non sono presenti PT-Handles o TP-handles. Inoltre, è S-coverable e sono presenti 22 S-components. Per questo net e per tutti quelli che verranno analizzati in seguito la possibilità di giungere al termine del processo è garantita e per questo motivo il Coverability graph coinciderà con il Reachability graph, esso è osservabile in Figura 10 e presenta 10 vertici e 14 archi.

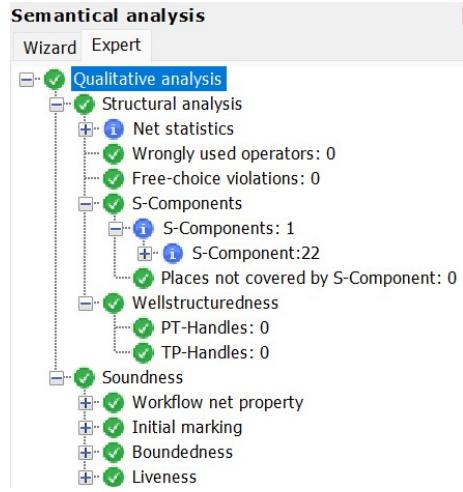


Fig. 2: Analisi semantica

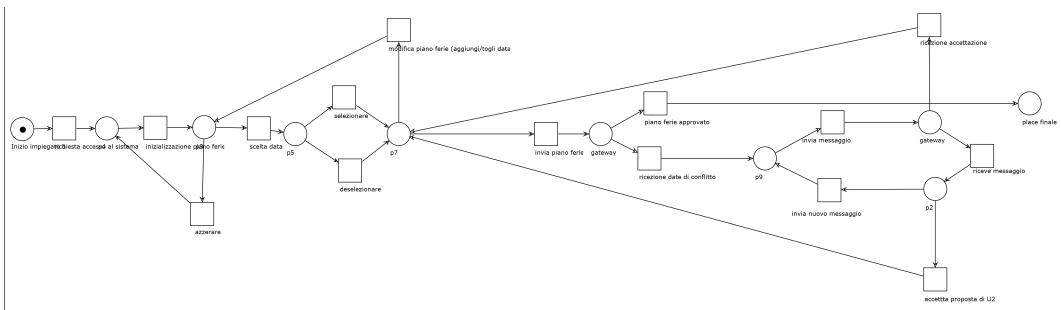


Fig. 3: Petri Net Impiegato 1

2.2 Impiegato 2

Il secondo impiegato manterrà lo stesso comportamento del primo nella parte di creazione del piano ferie, ma nel caso di conflitto, una volta ricevute le date, dovrà attendere un messaggio di proposta da parte dell'impiegato 1.

L'impiegato 2 potrà quindi decidere se accettare la proposta ricevuta oppure inviare

un messaggio di controproposta al collega.

Nel caso in cui decida di accettare la proposta tornerà al punto in cui sarà in grado di modificare il piano, se necessario, o inoltrarlo alla macchina per sottoporla a verifica.

Invece, nel caso in cui voglia rifiutare la proposta e di conseguenza inviare un messaggio di controproposta, l'impiegato 2 tornerà all'event-based gateway in attesa della notifica di accettazione della sua controproposta o di un nuovo messaggio da parte dell'impiegato 1 iniziando nuovamente il loop.

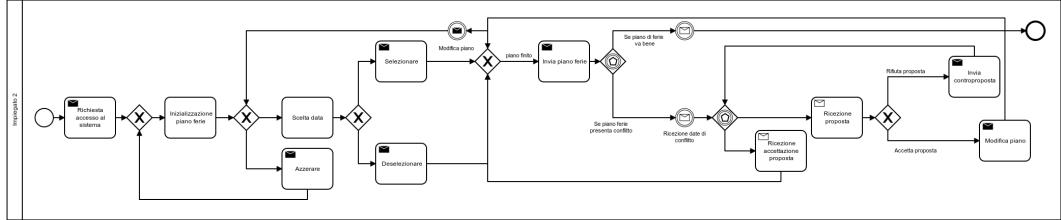


Fig. 4: BPMN Impiegato 2

In seguito, è stata effettuata la traduzione in Petri net, grazie alla quale sono stati ottenuti 9 places, 14 transizioni e 28 archi. Si tratta di un Workflow net in quanto presenta un place iniziale i ed uno finale o e ogni place e transizione appartiene ad un percorso tra i e o . La rete è S-net e rispetta tutte le proprietà della soundness, infatti non esistono dead task (come si può verificare nel coverability graph), e sono soddisfatte le proprietà di option to complete e proper completion. Come si può vedere dalla Figura 6 la rete è deadlock-free, bounded e free-choice, wellstructuredness, infatti non sono presenti PT-Handles o TP-handles. Inoltre, è S-coverable e sono presenti 23 S-components. Il Reachability graph, osservabile in Figura 10 presenta 9 vertici e 13 archi.

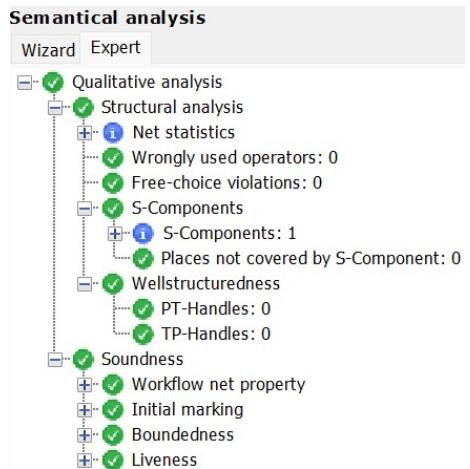


Fig. 5: Analisi semantica

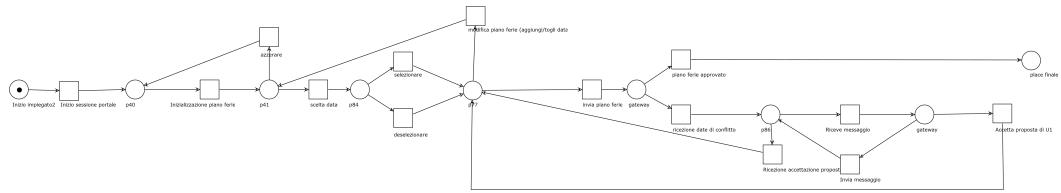


Fig. 6: Petri Net Impiegato 2

2.3 Sistema elettronico

Il sistema elettronico è la rappresentazione della piattaforma in cui gli impiegati possono creare i loro piani ferie e sottoporli ad approvazione. Una volta che gli impiegati effettuano il login, la macchina viene inizializzata ed avviene l'inizio della

sessione nel portale. La macchina agisce come interfaccia alle scelte degli impiegati, dopo l'inizio sessione può, tramite un event-based gateway, ricevere il task di memorizzare la selezione o deselezione di ciascun impiegato oppure azzerarne la memoria temporanea. Dopo ogni memorizzazione, la macchina viene condotta ad un altro event-based gateway, dove in base al task scelto dall'impiegato, può ricevere, da parte dell'impiegato, nel primo caso la richiesta di aggiungere modifiche al piano ferie e quindi il ritorno da parte della macchina al primo gateway per memorizzare ulteriori date, oppure, nel secondo caso, se l'utente decide di sottoporre il suo piano ferie ad approvazione, la macchina ottiene direttamente il piano ferie e procede alla verifica.

Una volta ricevuti i piani ferie compilati dagli attori coinvolti, il sistema ne effettuerà la verifica. Nel caso in cui i piani ferie siano conformi l'un l'altro, ossia ogni data selezionata non collida con quelle selezionate dal collega, il sistema approverà i piani ferie inviando una notifica a entrambi e salvando i piani nel database della piattaforma.

Se invece nei piani inoltrati sono presenti date coincidenti, il sistema rileverà le date di conflitto e le inoltrerà agli utenti. La piattaforma rimarrà in attesa del messaggio inviato dall'impiegato 1 all'impiegato 2. Una volta inoltrato il messaggio, il sistema, tramite un event-based gateway, rimarrà in attesa della decisione del secondo impiegato, il quale prendendo la sua decisione può generare due differenti percorsi:

- Nel caso in cui decidesse di accettare la proposta appena ricevuta, la piattaforma riceverà la notifica di accettazione e la inoltrerà all'impiegato 1, tornando quindi all'event-based-gateway in cui l'impiegato può scegliere se modificare il piano ferie o inviare il nuovo piano ferie da verificare.
- Nel caso in cui l'impiegato 2 non accettasse la proposta ricevuta e decidesse di rispondere con un messaggio, la piattaforma riceverà il messaggio e lo inoltrerà all'impiegato 1. A questo punto la piattaforma giunge ad un altro event-based gateway dove attenderà un ulteriore messaggio da parte dell'impiegato 1 oppure la notifica di accettazione della proposta ricevuta. Nel primo caso, la ricezione di un nuovo messaggio riconduce all'inizio del ciclo di messaggistica. Questo garantisce la possibilità di continuare la conversazione finché gli impiegati non avranno trovato un accordo. Nel secondo caso, la piattaforma inoltra la notifica di accettazione e ritorna all'event-based gateway iniziale aspettando la decisione di modificare il piano ferie o la richiesta di approvazione del piano ferie di uno o entrambi gli impiegati a seconda dell'accordo preso tra i due.

La macchina avrà quindi un ruolo attivo soltanto nella verifica dei piani ferie, mentre sarà sempre passiva per assecondare le azioni degli impiegati, lo si può notare dal fatto che siano presenti esclusivamente event-based gateway a parte lo XOR-join iniziale che permette agli utenti di tornare indietro e lo XOR-split in cui verifica i piani ferie.

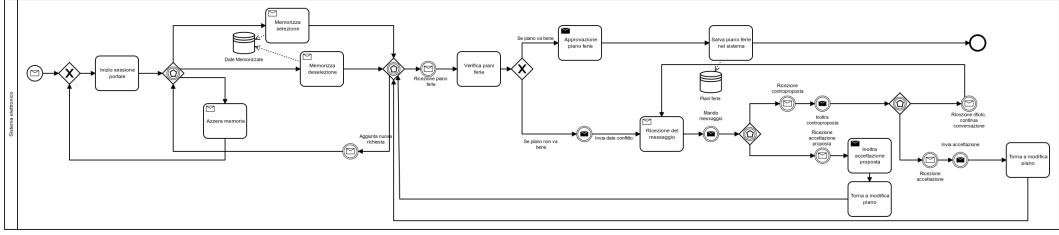


Fig. 7: BPMN Sistema Elettronico

In seguito, è stata effettuata la traduzione in Petri net, grazie alla quale sono stati ottenuti 17 places, 22 transizioni e 44 archi. Si tratta di un Workflow net in quanto presenta un place iniziale i ed uno finale o e ogni place e transizione appartiene ad un percorso tra i e o . La rete è S-net e rispetta tutte le proprietà della soundness, infatti non esistono dead task (come si può verificare nel coverability graph), e sono soddisfatte le proprietà di option to complete e proper completion. Come si può vedere dalla Figura 9 la rete è deadlock-free, bounded e free-choice, wellstructuredness, infatti non sono presenti PT-Handles o TP-handles. Inoltre, è S-coverable e sono presenti 39 S-components. Il Reachability graph, osservabile in Figura 10 presenta 18 vertici e 22 archi.

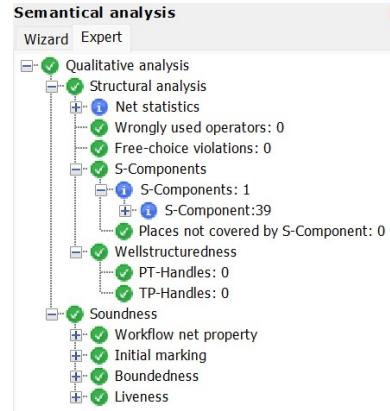


Fig. 8: Analisi semantica

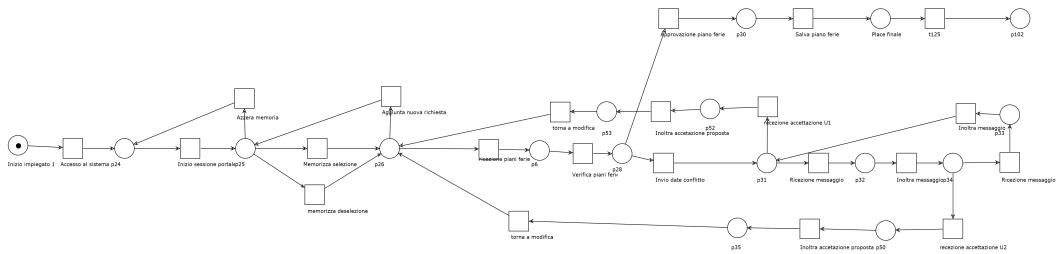


Fig. 9: Petri Net Sistema Elettronico

2.4 Reachability graphs

Di seguito sono illustrati i Reachability graph rappresentanti i tre attori. Tramite i grafici è possibile verificare le proprietà della soundness, in quanto c'è sempre la possibilità di giungere al termine del processo e con un solo token nel place finale (proper completion), inoltre non ci sono dead task.

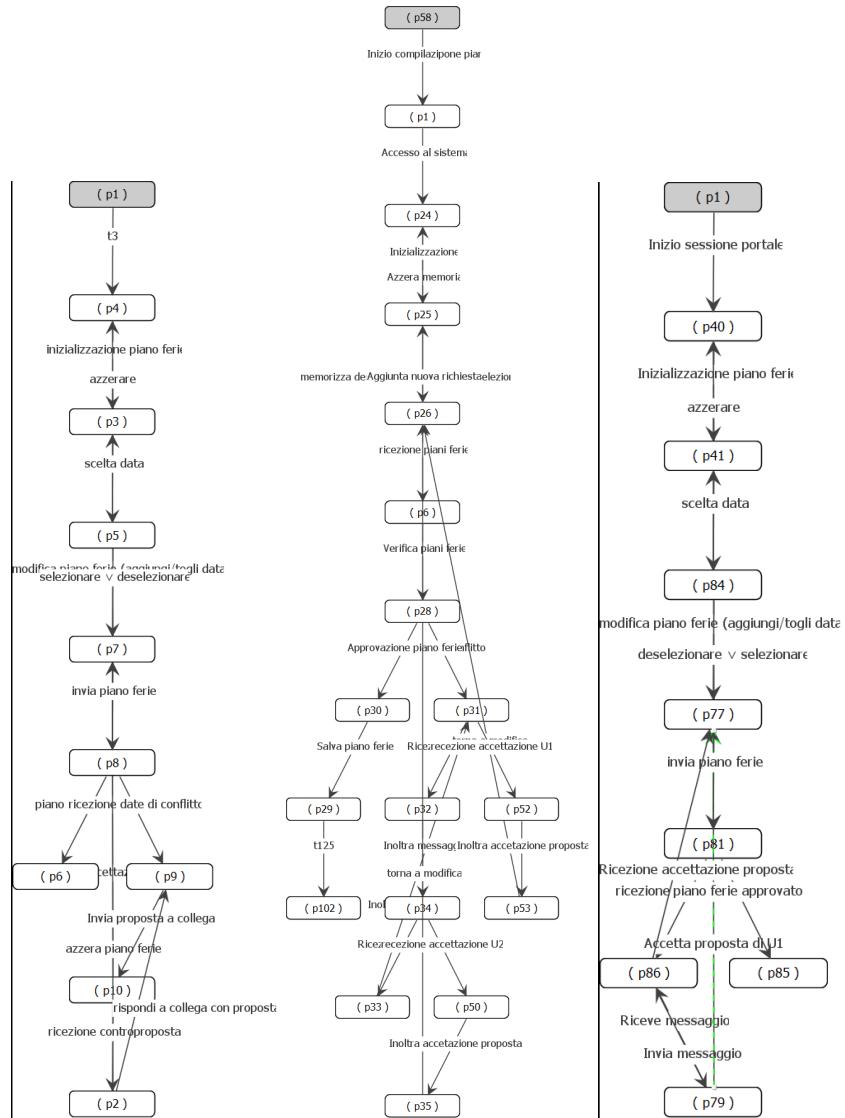


Fig. 10: Reachability graph: Impiegato 1, Sistema elettronico, Impiegato 2

3 Collaborazione tra gli agenti

3.1 BPMN

Rappresentati i singoli workflow modules e i relativi comportamenti verrà ora illustrata l'interazione tra di loro, ottenendo in questo modo un workflow system grazie alle interfacce che permettono di favorire la collaborazione tra agenti.

Per ottenere il diagramma BPMN visualizzabile in Figura 12 sono stati selezionati i singoli diagrammi creati in precedenza per ogni attore e combinati tra loro per creare l'interazione. I collegamenti per favorire la comunicazione tra gli attori sono stati compiuti tramite l'uso di interfacce (message flow arrows) per favorire il message-control flow. In particolare, le interfacce sono state aggiunte ogni qualvolta l'utente decida di compiere un'azione nei confronti del sistema come ad esempio: eseguire l'accesso al sistema, selezionare/deselezionare una data, azzerare la selezione, chiedere l'approvazione del piano ferie, ecc. e viceversa quando il sistema deve comunicare con gli impiegati, come il caso in cui debba riferire loro l'esito della verifica del loro piano ferie.

3.2 Petri Net

Per quanto riguarda la conversione in Petri Net (effettuata tramite l'utilizzo di WoPeD), per rendere il diagramma un Worflow net, non è bastato unire le reti dei singoli attori, ma è divenuto essenziale rispettare la condizione necessaria dei Workflow net ossia di avere un place iniziale e un place finale, i quali sono stati creati in questo modo:

- I place iniziali dei singoli attori sono stati unificati in un AND Start per il place iniziale in quanto devono attivarsi entrambi i flussi contemporaneamente.
- AND End per fare in modo che entrambi i flussi arrivino alla fine, quindi finché non arriveranno i tokens da entrambi il sistema non potrà finire. Infatti, finché non si sarà trovato un accordo tra i due impiegati tramite il sistema, la rete non potrà giungere ad una conclusione.

In Figura 13 è possibile vedere la Petri Net completa.

Per effettuare l'analisi semantica non è stato possibile utilizzare WoPeD in quanto il caricamento della rete era computazionalmente troppo costoso e non giungeva ad un risultato, per questo motivo è stato utilizzato Woflan per ottenere comunque delle conferme sul funzionamento della rete. Si può osservare come tutti i task siano live e il sistema sia sound.

In Figura 14 è rappresentato il Reachability graph rappresentante l'intero processo.

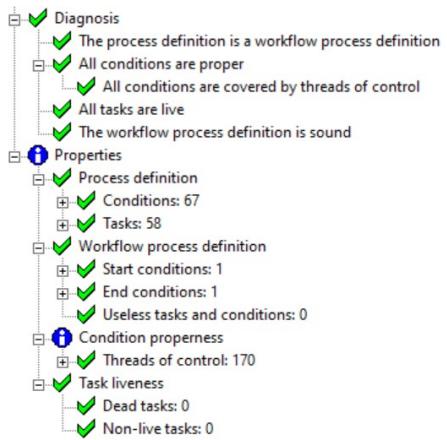


Fig. 11: Analisi semantica

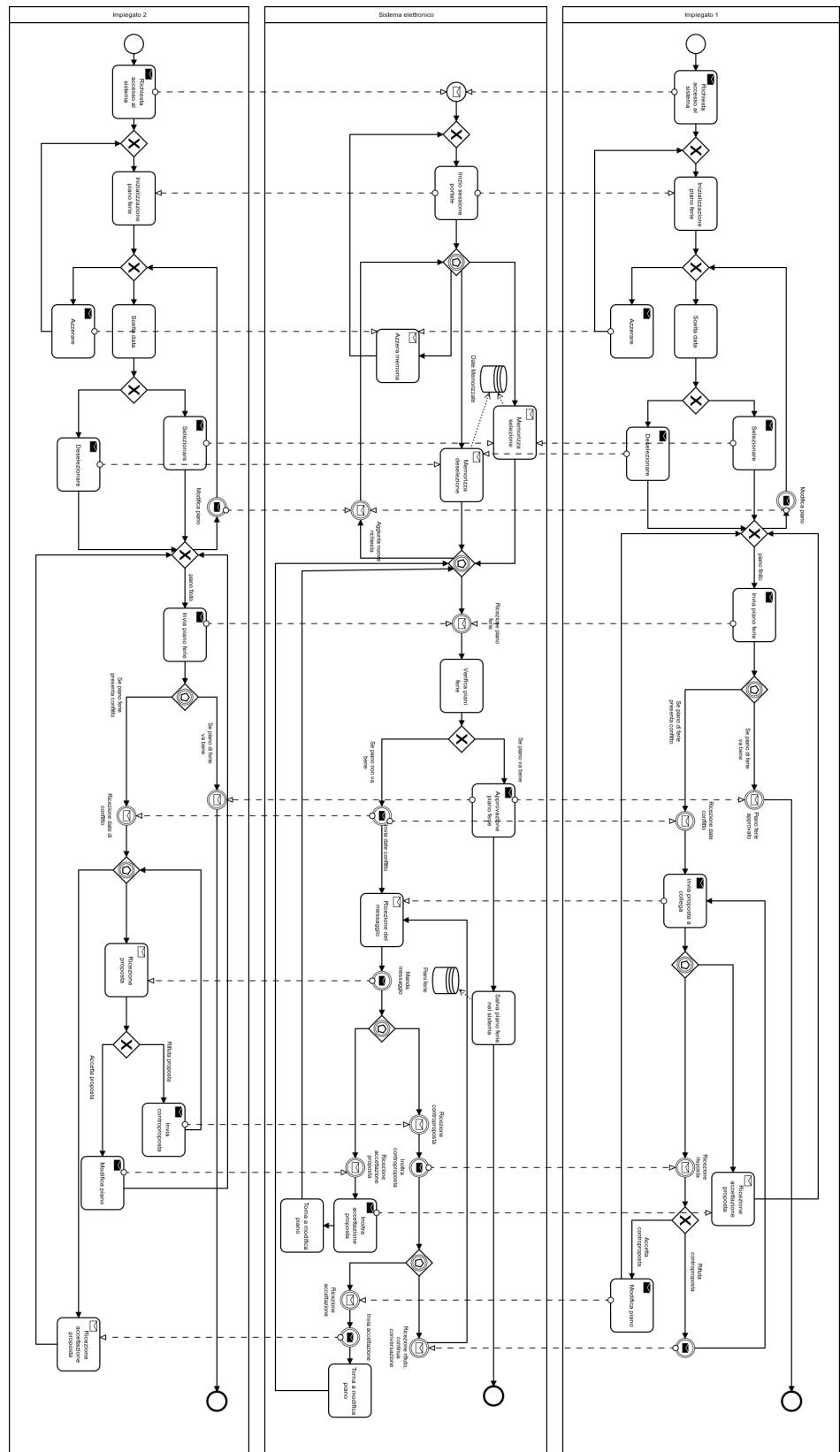


Fig. 12: Diagramma BPMN completo

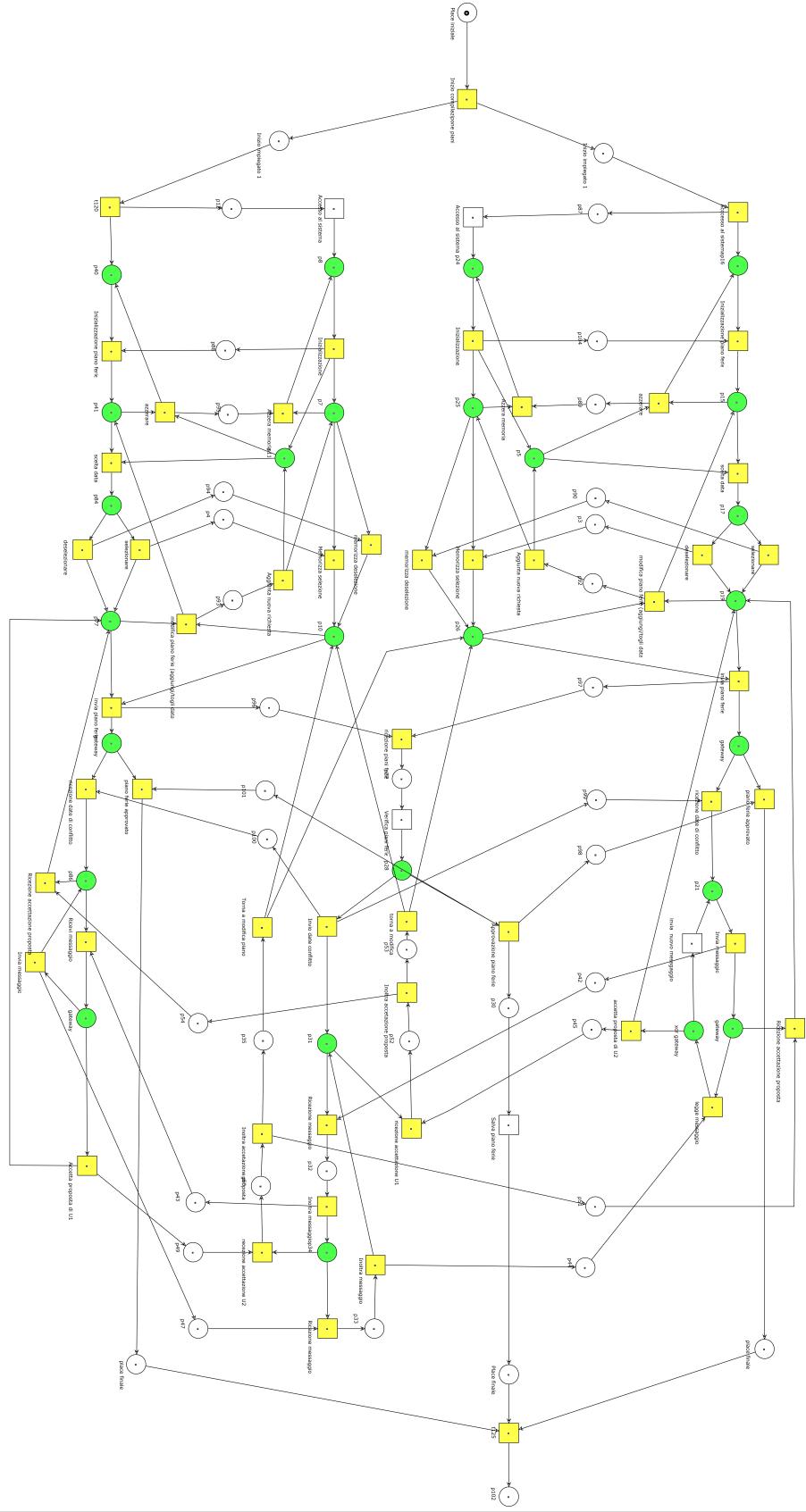


Fig. 13: Petri Net completa

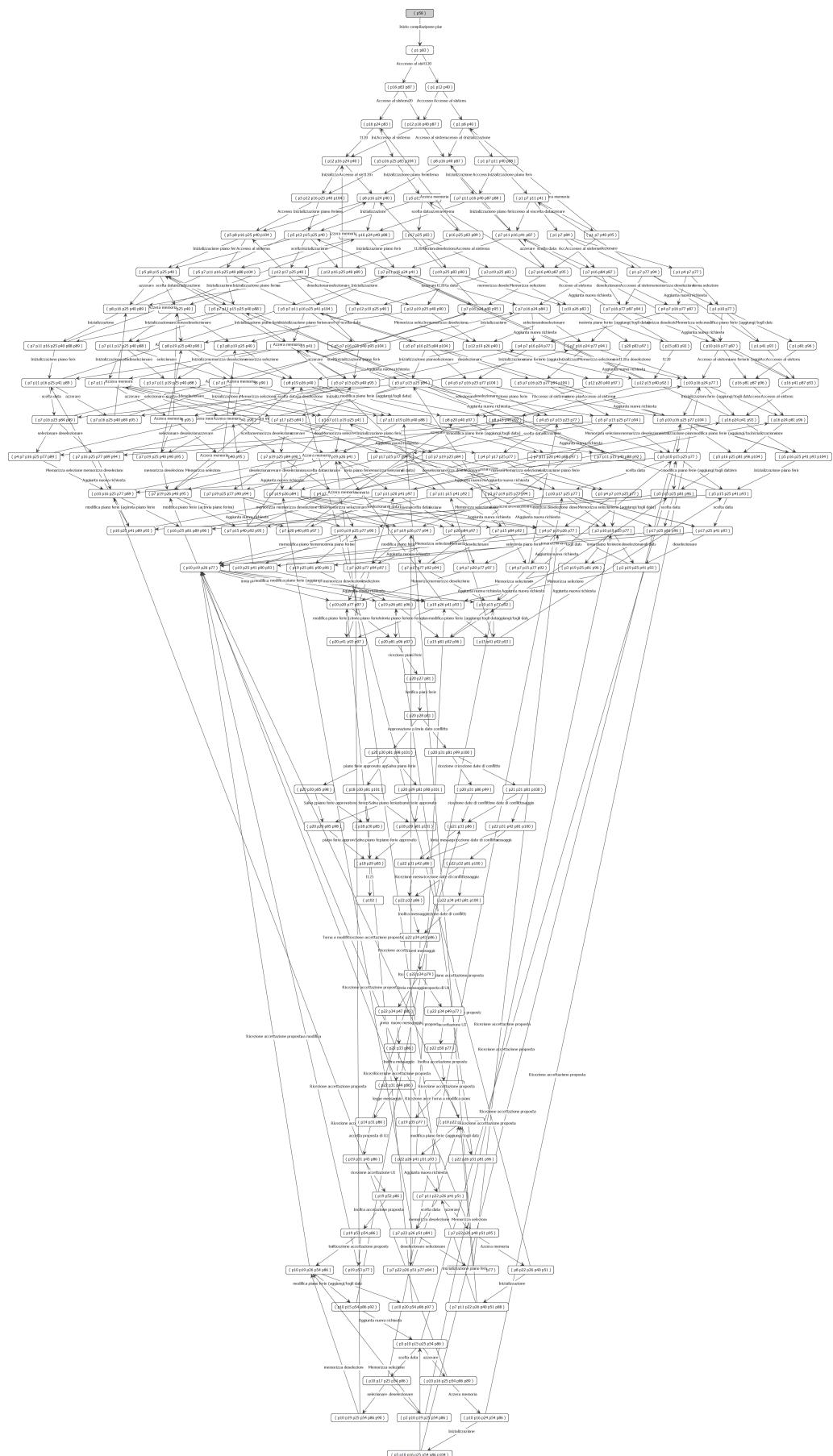


Fig. 14: Reachability Graph completo

4 Caso Alternativo

4.1 BPMN

Infine, i processi sono stati modificati in modo che i conflitti vengano segnalati dal sistema prima dell'approvazione finale. Per ottenere questa struttura è stato aggiunto un controllo da parte del sistema ogni qualvolta un utente seleziona una nuova data.

Per l'utente il cambiamento è dato dal fatto che, una volta selezionata una data, egli dovrà attendere che il sistema controlli che quel determinato giorno non sia già stato selezionato in precedenza dall'altro impiegato. Nel caso in cui non ci sia un conflitto, potrà continuare come in precedenza, ossia avrà la possibilità di selezionare o deselectare altre date, azzerare il piano, oppure inviare il piano ferie per sottoporlo a controllo. In caso contrario, il sistema non gli darà la possibilità di aggiungere quella data e lo riporterà alla possibilità di scegliere una nuova data o azzerare il piano. L'unica variazione è rappresentata da un event-based gateway che attende una risposta del problema in base alla presenza o meno di un conflitto nella data selezionata. In seguito, non sarà necessaria la messaggistica tra impiegati in quanto il piano sarà già privo di conflitti e il piano verrà direttamente approvato. Il sistema, invece, prima di memorizzare la selezione dell'impiegato, dovrà verificare che la data non sia presente tra quelle già selezionate dal collega. Per quanto riguarda la deselectazione e l'azzeramento, invece, non ci sono variazioni. In seguito, l'event-based gateway aspetta che entrambi gli utenti abbiano finito di compilare i piani, li approva e infine li salva nel database.

4.2 Petri Net

Successivamente, dal BPMN è stata effettuata la trasformazione in Petri net visualizzabile in figura 17 tramite le regole di traduzione. Con WoPeD è stata effettuata l'analisi semantica grazie alla quale sono stati ottenuti 55 places, 47 transizioni e 144 archi.

Si tratta di un Workflow net in quanto presenta un place iniziale e uno finale ed esiste un percorso tra loro, e soddisfa tutte le proprietà che permettono di definire la rete sound, infatti è bounded e live. Come si può vedere dalla Figura la rete presenta 4 violazioni della free-choice e 160 PT-Handles e 144 TP-Handles, quindi non è wellstructuredness.

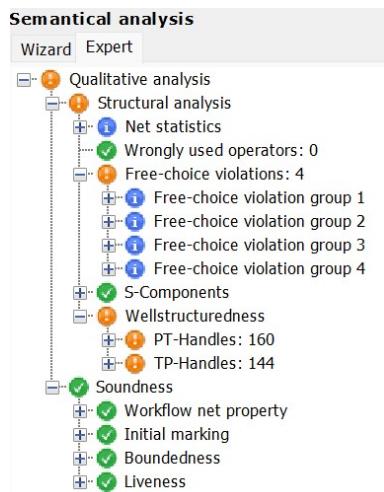


Fig. 15: Analisi semantica

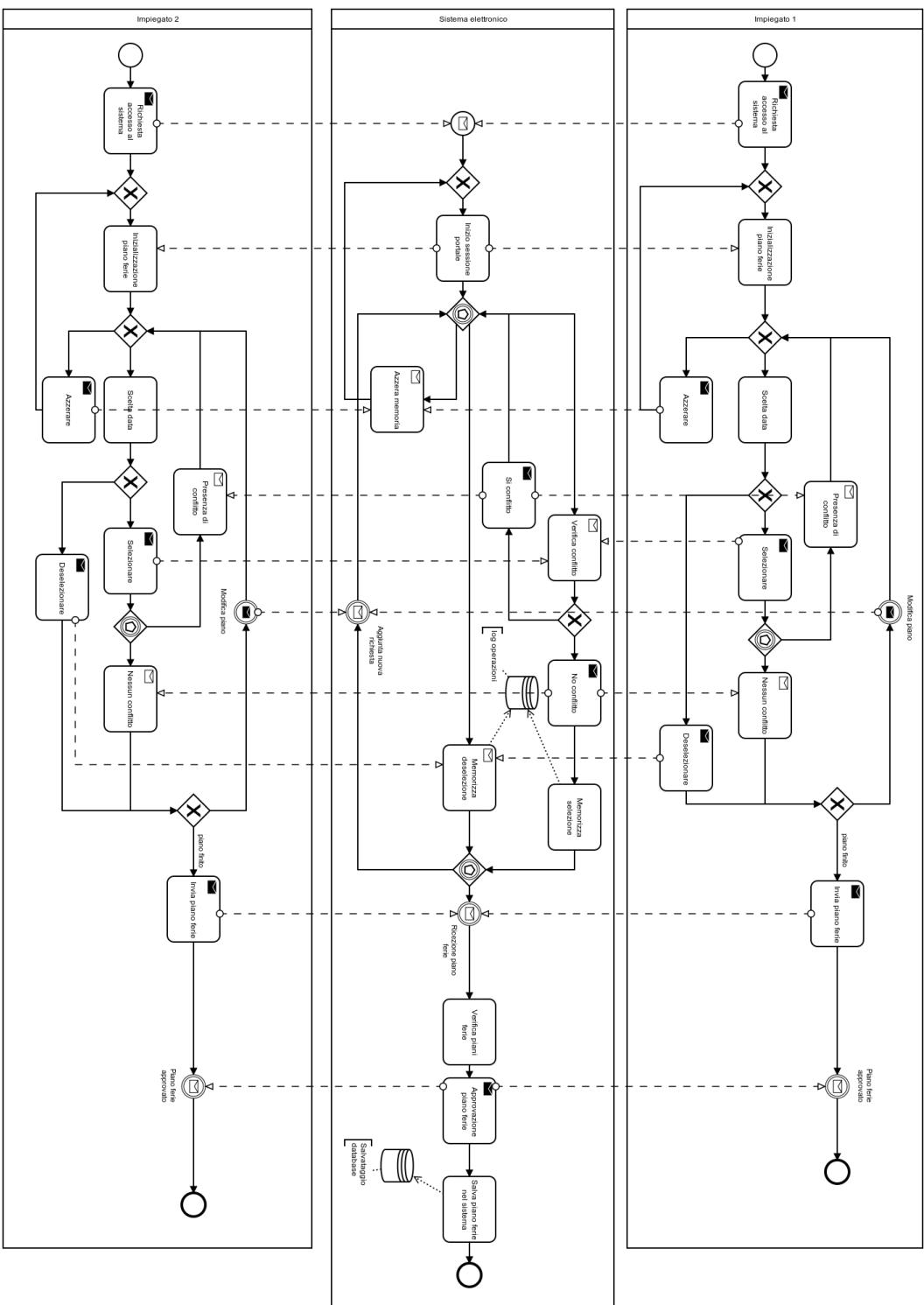


Fig. 16: BPMN modificato

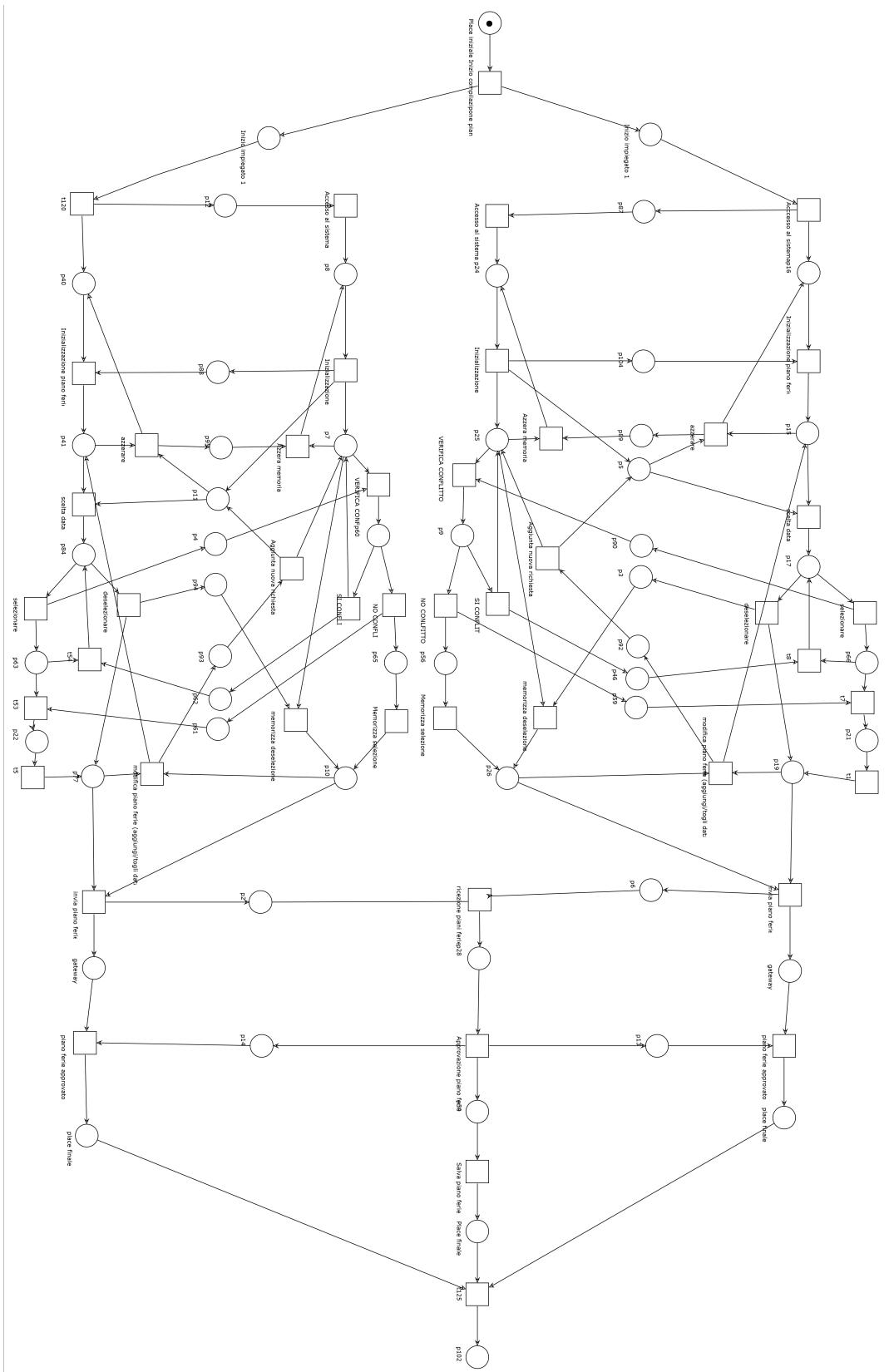


Fig. 17: Petri Net modificata