



**POLITECNICO**  
MILANO 1863

# Sistemi Informativi (per il settore dell'informazione)

Ingegneria Informatica

BPMN

Business Process Model and Notation

# Concetti avanzati

Collaborazione

Dati

Argomento non  
parte di questo  
corso

Transazioni

Argomento non  
parte di questo  
corso

# Obiettivi

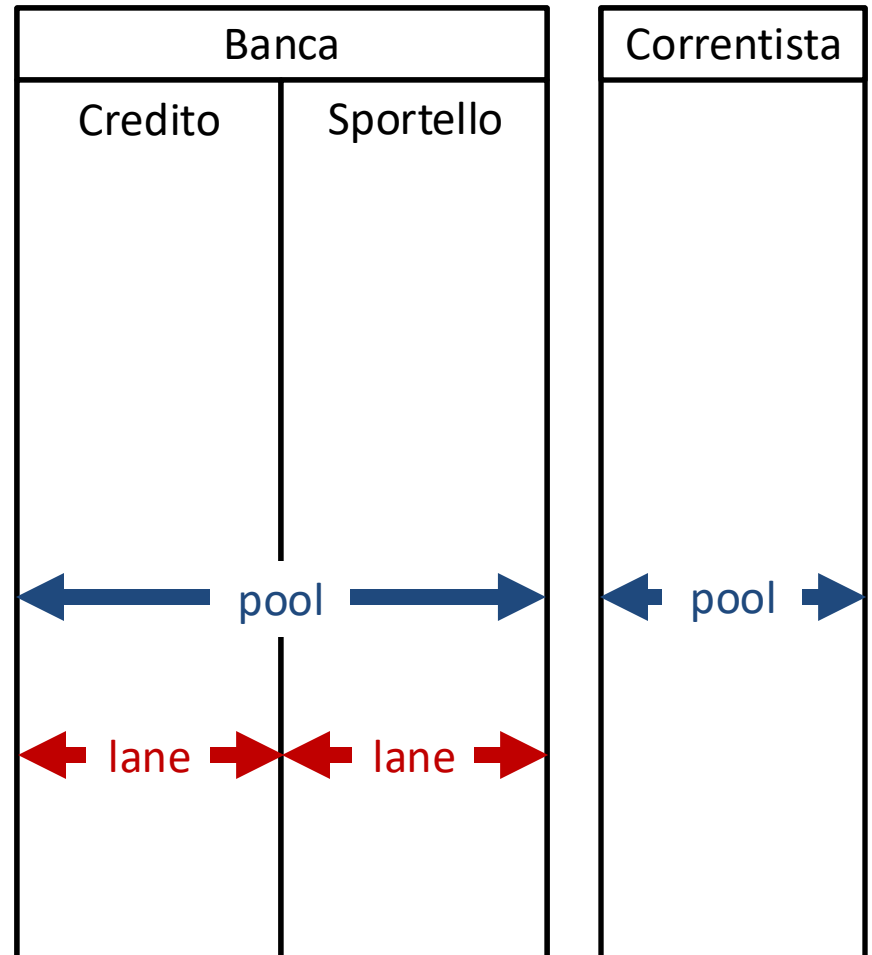
- Descrivere l'interazione tra diversi processi, gestiti da diverse entità detti “partecipanti”, ovvero “orchestratori”, sistemi che possono gestire l'esecuzione di un modello di processo
- Le entità possono essere
  - Diverse organizzazioni
  - Diverse divisioni all'interno della medesima organizzazione
- L'interazione è governata e definita da uno scambio di messaggi

# Pool

- Elemento grafico che rappresenta un partecipante
  - All'interno contiene un processo orchestrato gestito interamente dal partecipante (es. un'organizzazione)
- Può essere:
  - Black box: non si danno i dettagli del processo interno, ma solo lo scambio informativo con gli altri pool (è un segnaposto di «dialogo con altro orchestratore»)
  - White box: sono presenti i dettagli del processo interno, o un sottoinsieme dei dettagli, e i messaggi sono usati per definire come sono gestiti gli scambi informativi

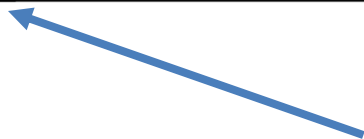
# Pool e lane

- Il contesto è importante
- Più partecipanti tra loro **autonomi**: *pool*
- Più attori per ogni partecipante: *lane*
- Un evento terminate ha effetto solo all'interno di un pool



# Black box pool

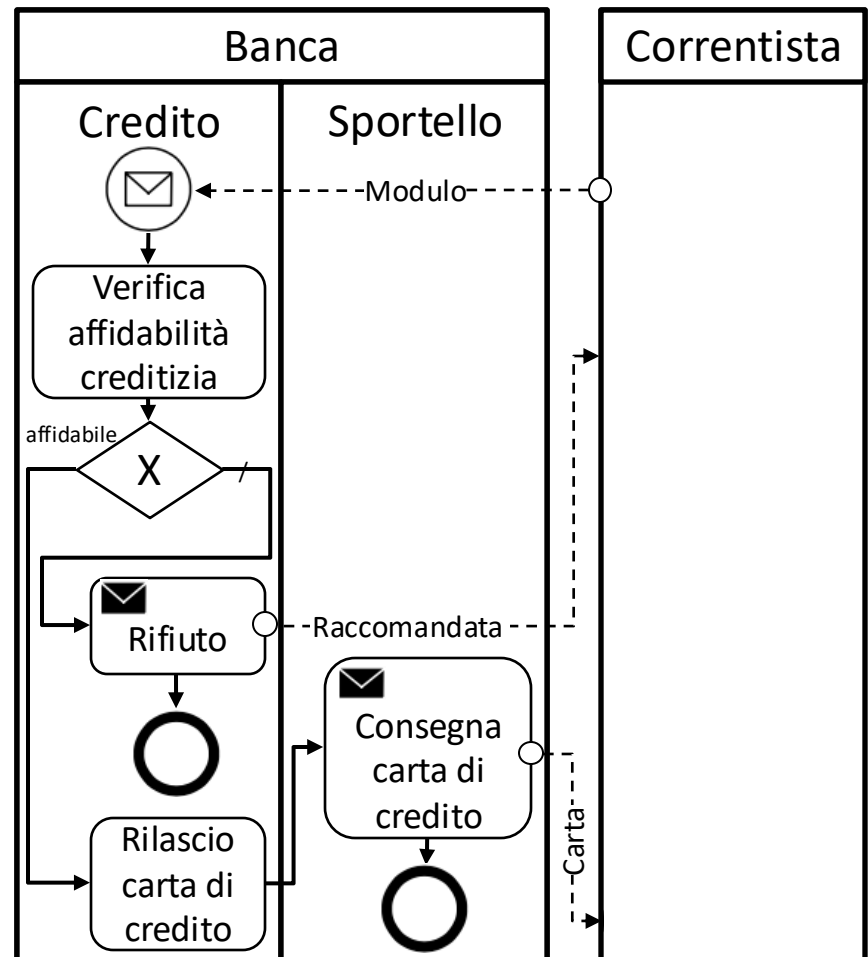
Name	
------	--



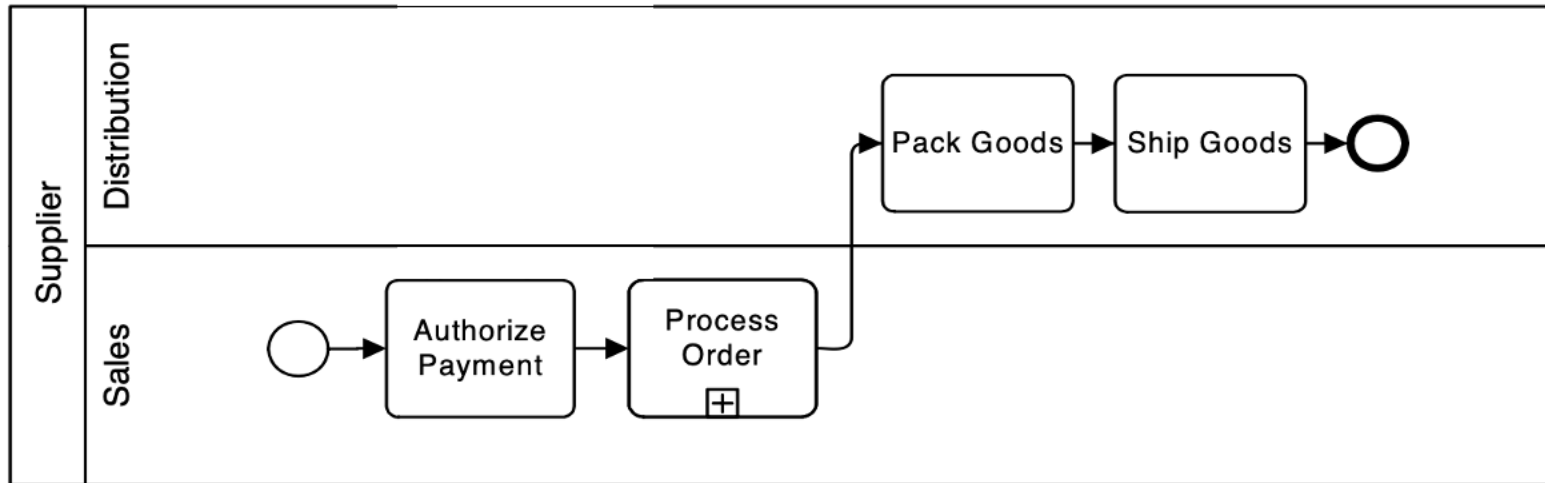
Nome del partecipante

# Black box pool

- Partecipanti a «scatola nera»: noto il solo flusso di messaggi.



# White box pool



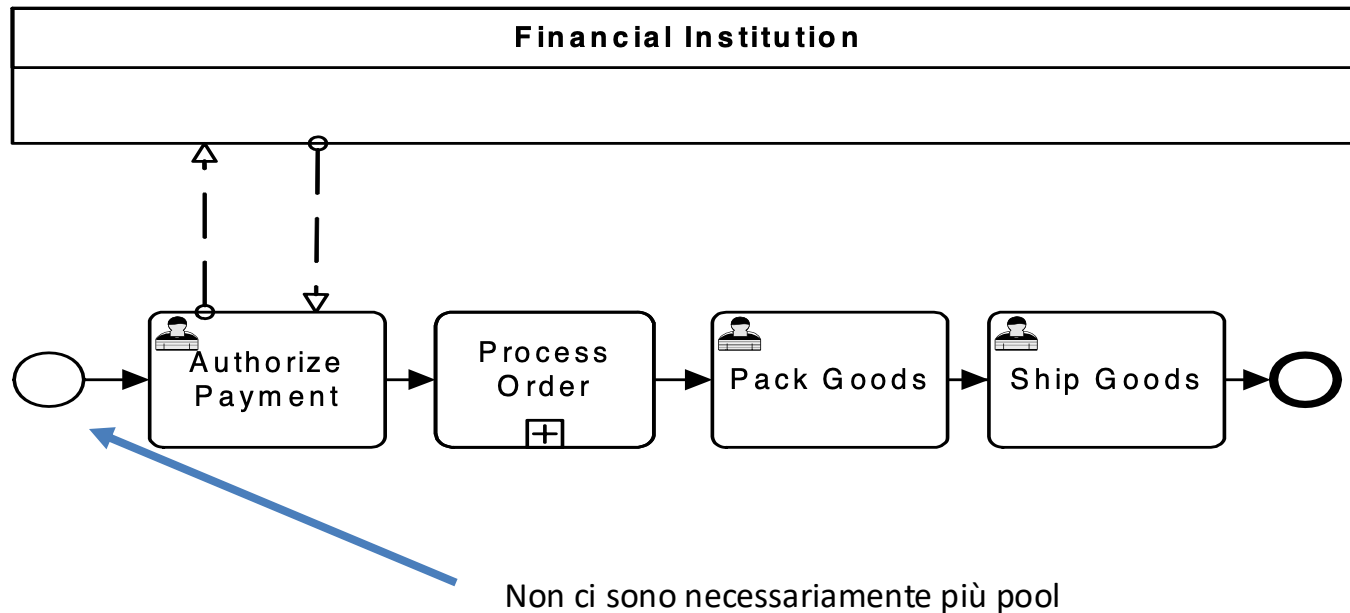
Partecipanti a «scatola bianca»:  
noti flussi di attività e messaggi.



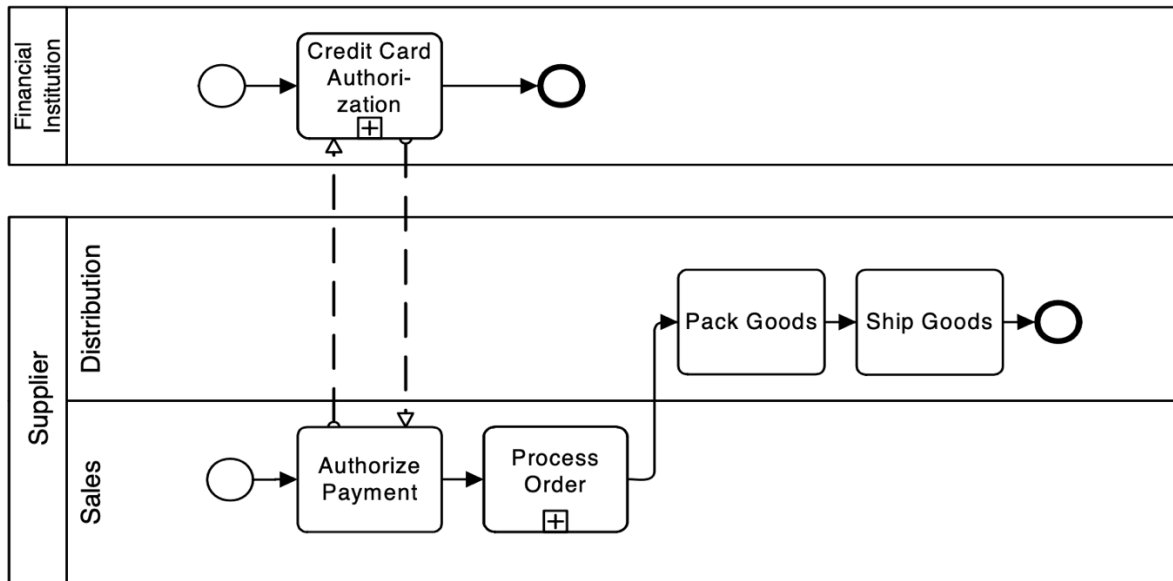
# BPMN Collaboration diagram

- È un BPMN diagram dove sono presenti uno o più pool e il protocollo che governa la loro interazione è definita in termini di messaggi scambiati
  - Non sono necessariamente messaggi digitali, potrebbero anche messaggi fisici od oggetti (anche se l'icona è sempre la «busta»)
- È possibile descrivere la collaboration tra due o più collaborazioni secondo:
  - Il punto di vista specifico di uno dei partecipanti
  - Un punto di vista neutro rispetto ai partecipanti

# Collaboration diagram secondo un punto di vista



# Collaboration diagram neutro



# Collaborazione ed eventi



- Generico
  - Nessuna specifica
- Messaggio
  - Accetta UNO E UN SOLO flusso di messaggi in ingresso\uscita
- Segnale
  - Visibile da ogni pool
- Tempo
  - Nessuna specifica
- Condizione
  - Nessuna specifica
- Terminazione
  - Agisce SOLO sulla POOL in cui è presente
- Errore
  - Nessuna specifica

# Scambi di messaggi

- Fra partecipanti (pool):



*flusso di attività*



*flusso di messaggi*

- Fra attori (lane):

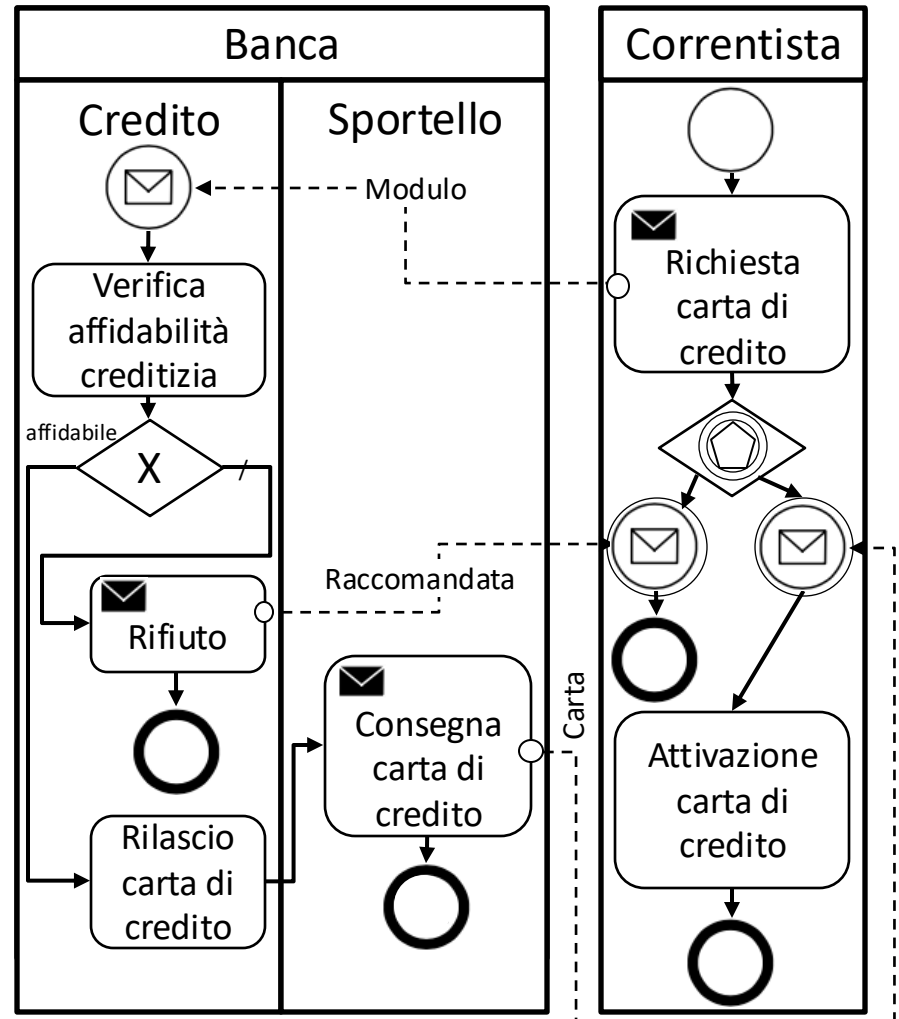


*flusso di attività*

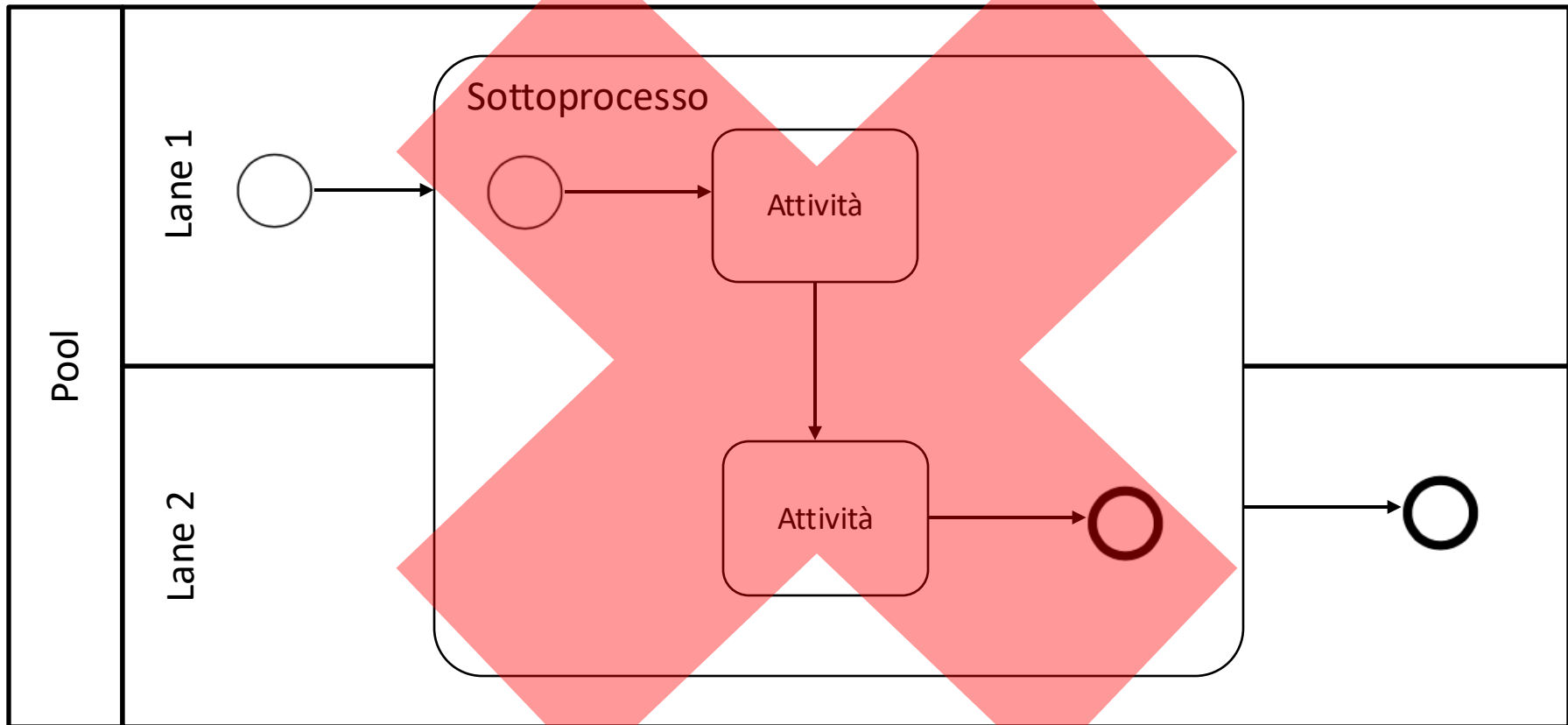


*flusso di messaggi*

: *very bad*



# Collaborazione e sottoprocessi

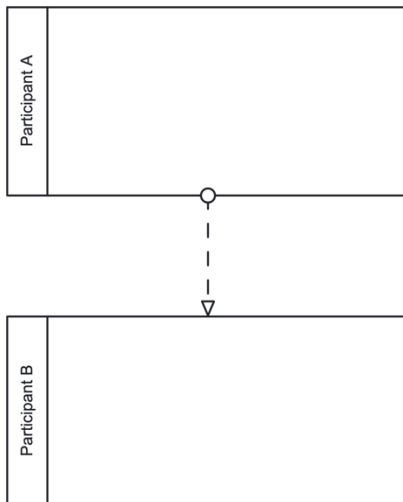


❌ No sottoprocessi, attività o eventi a cavallo di più lane o pool ❌  
(very bad)

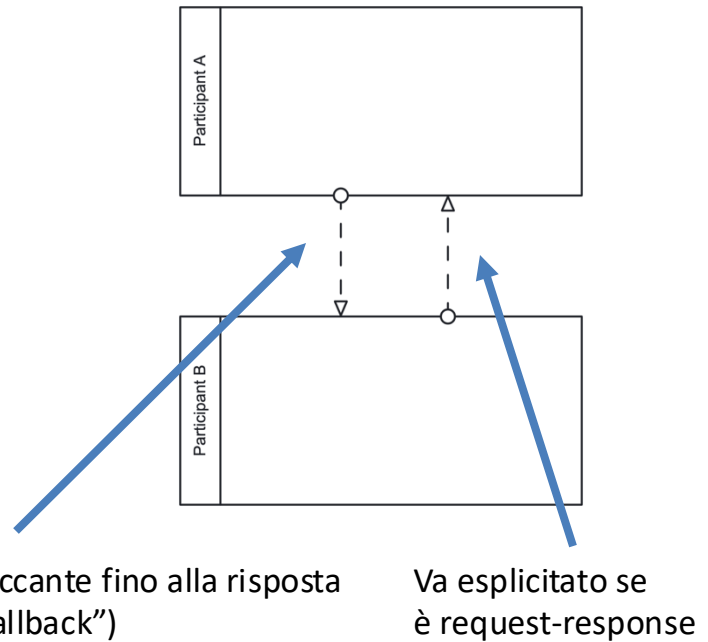
# Pattern scambio messaggi

I messaggi definiscono le interazioni tra partecipanti secondo due pattern principali

## One-way

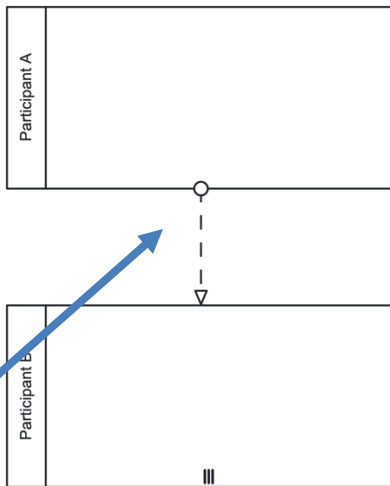


## Request-response



# Multi-instance pool

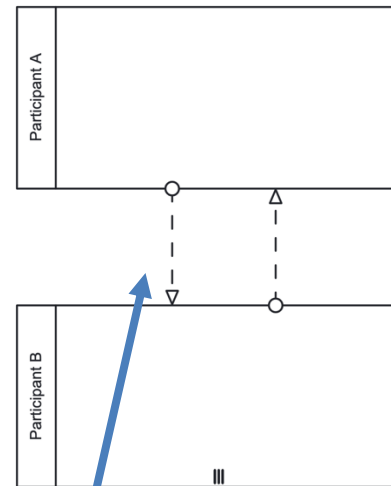
## One-way



Messaggio broadcast  
alle istanze esistenti

Classe di partecipanti: A dialoga  
con uno o più partecipanti B,  
identici ai fini esterni

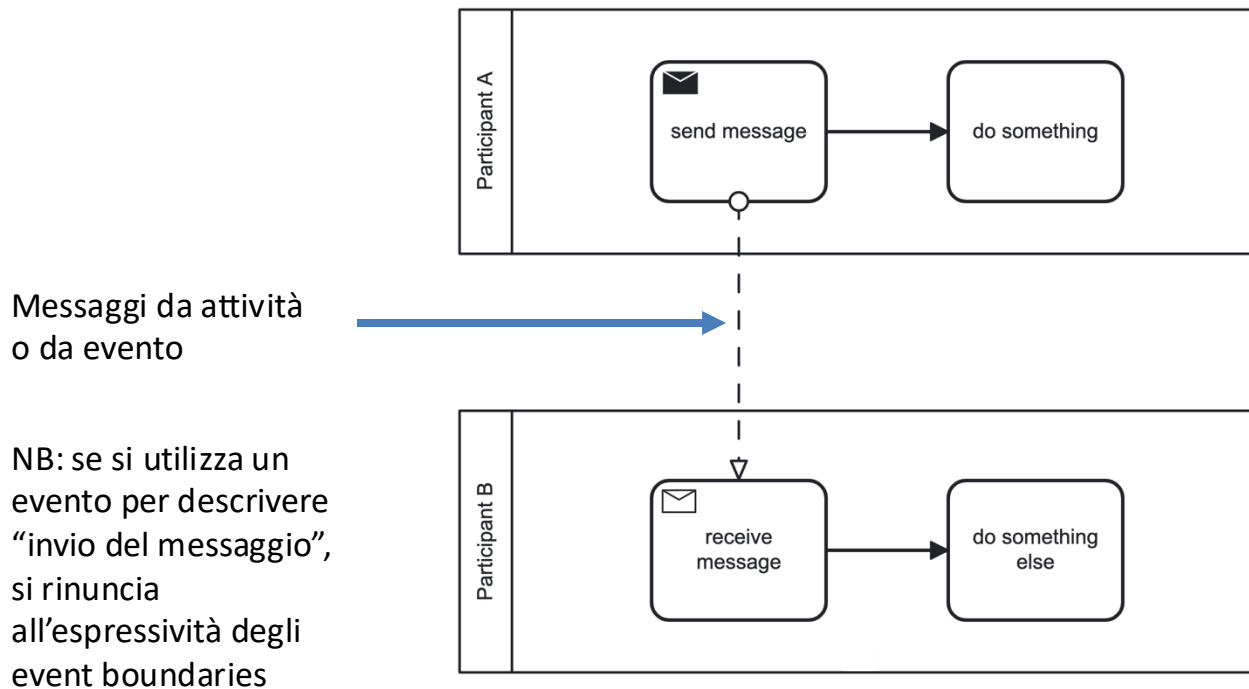
## Request-response



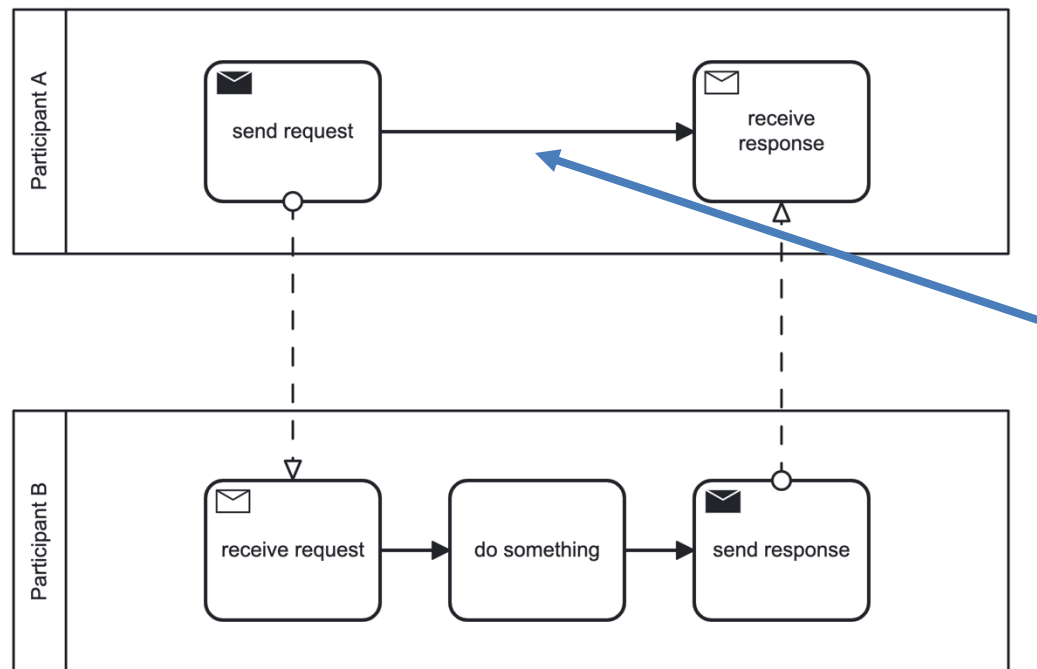
Attenzione alla sincronizzazione  
tra istanze differenti



# Message task – one-way

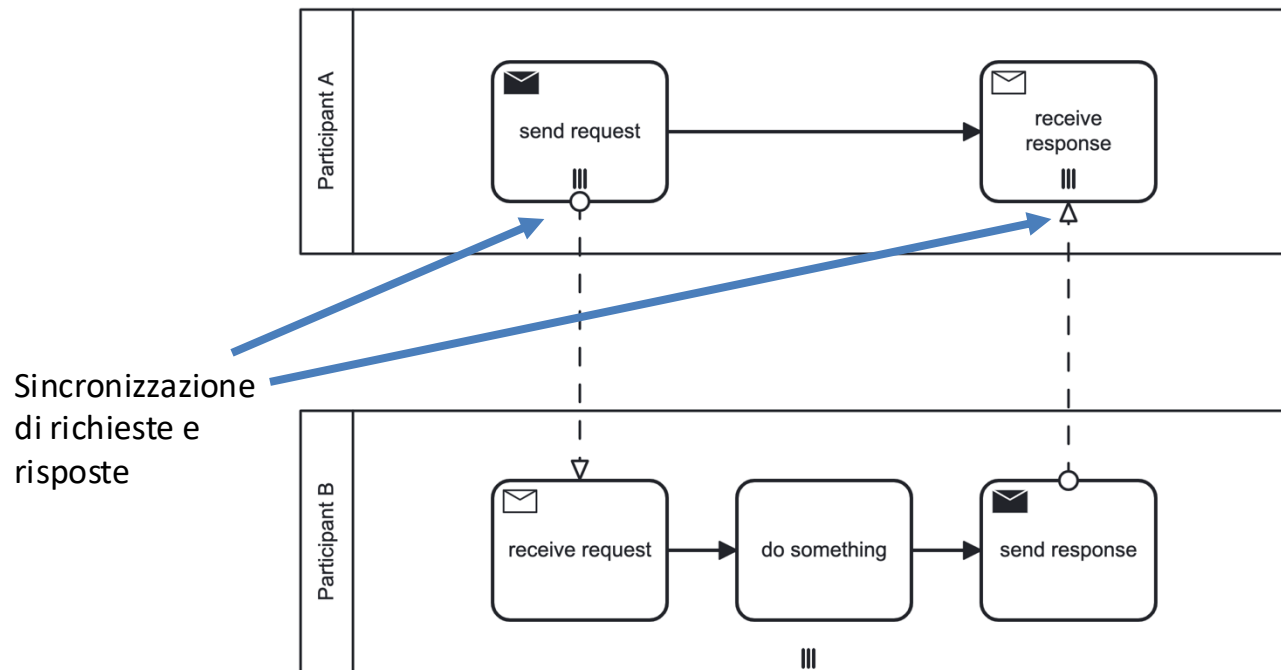


# Message task – request-response



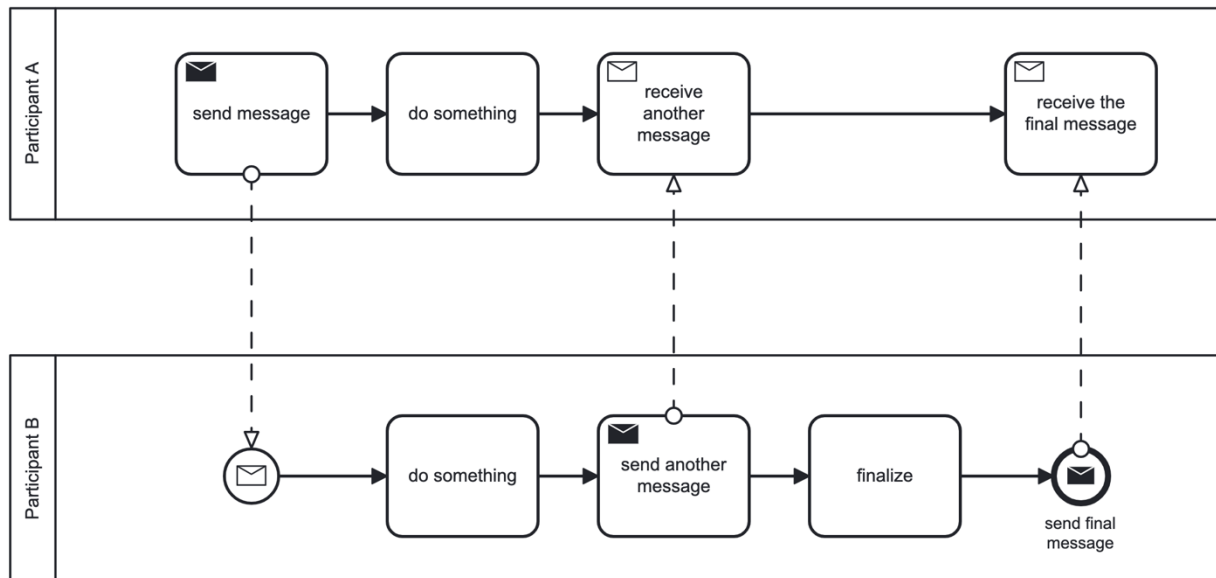
Se siamo nel caso di una request-response, non può esserci altro tra le due attività, caso bloccante (se c'è, siamo nel caso di doppio one way)

# Multi-instance pool

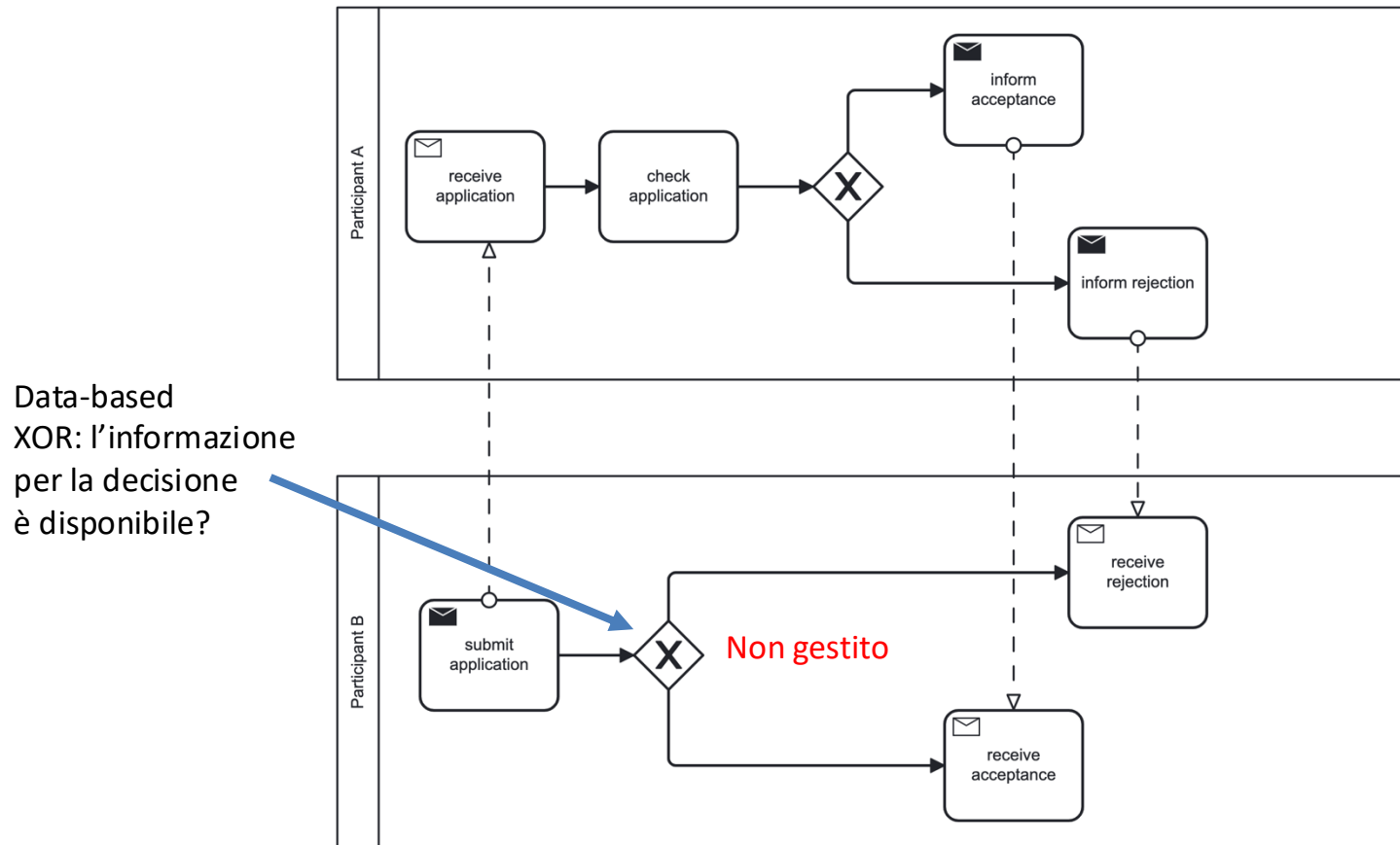


Come vengono allineate le singole richieste e risposte?

# Combinazione di messaggi: serie

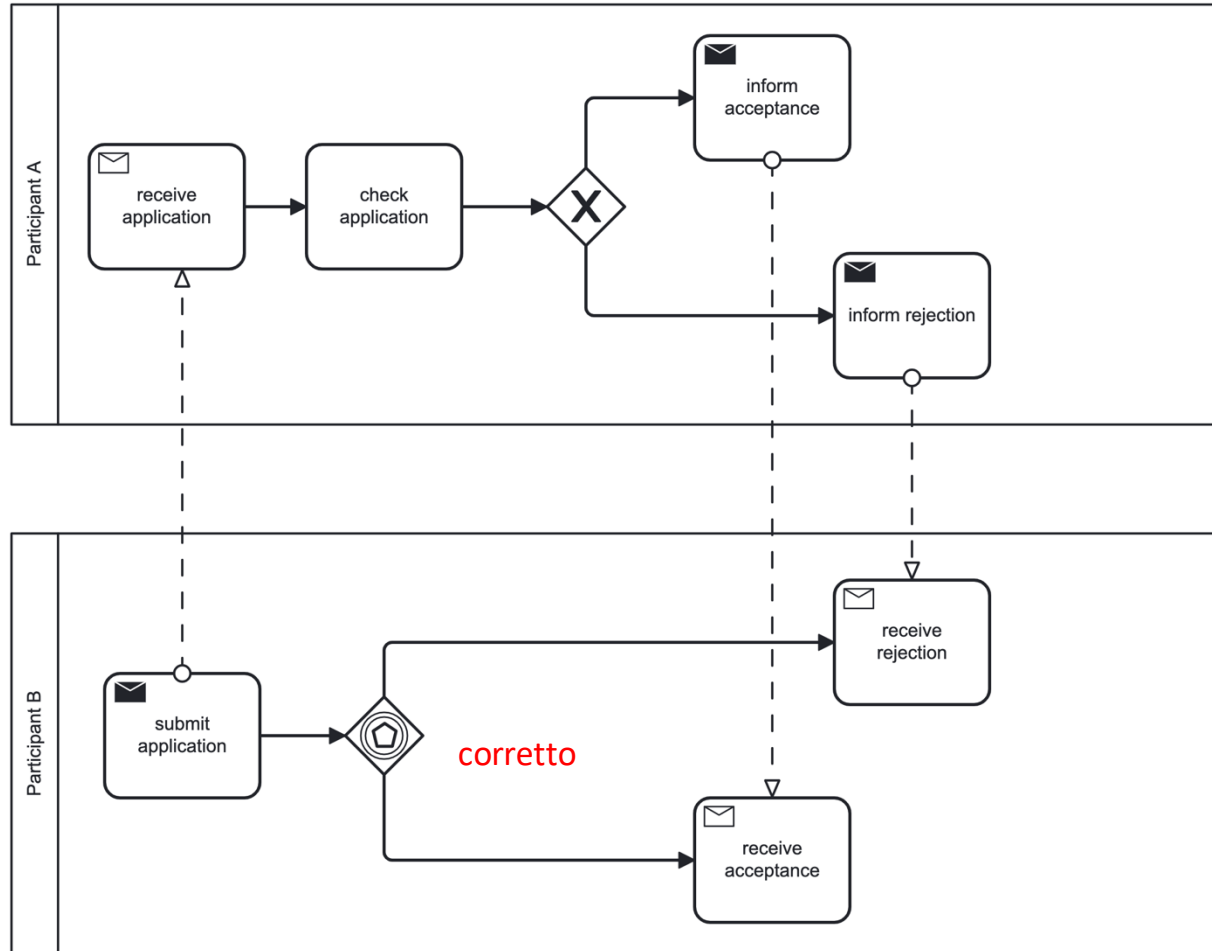


# Attenzione ai punti di decisione

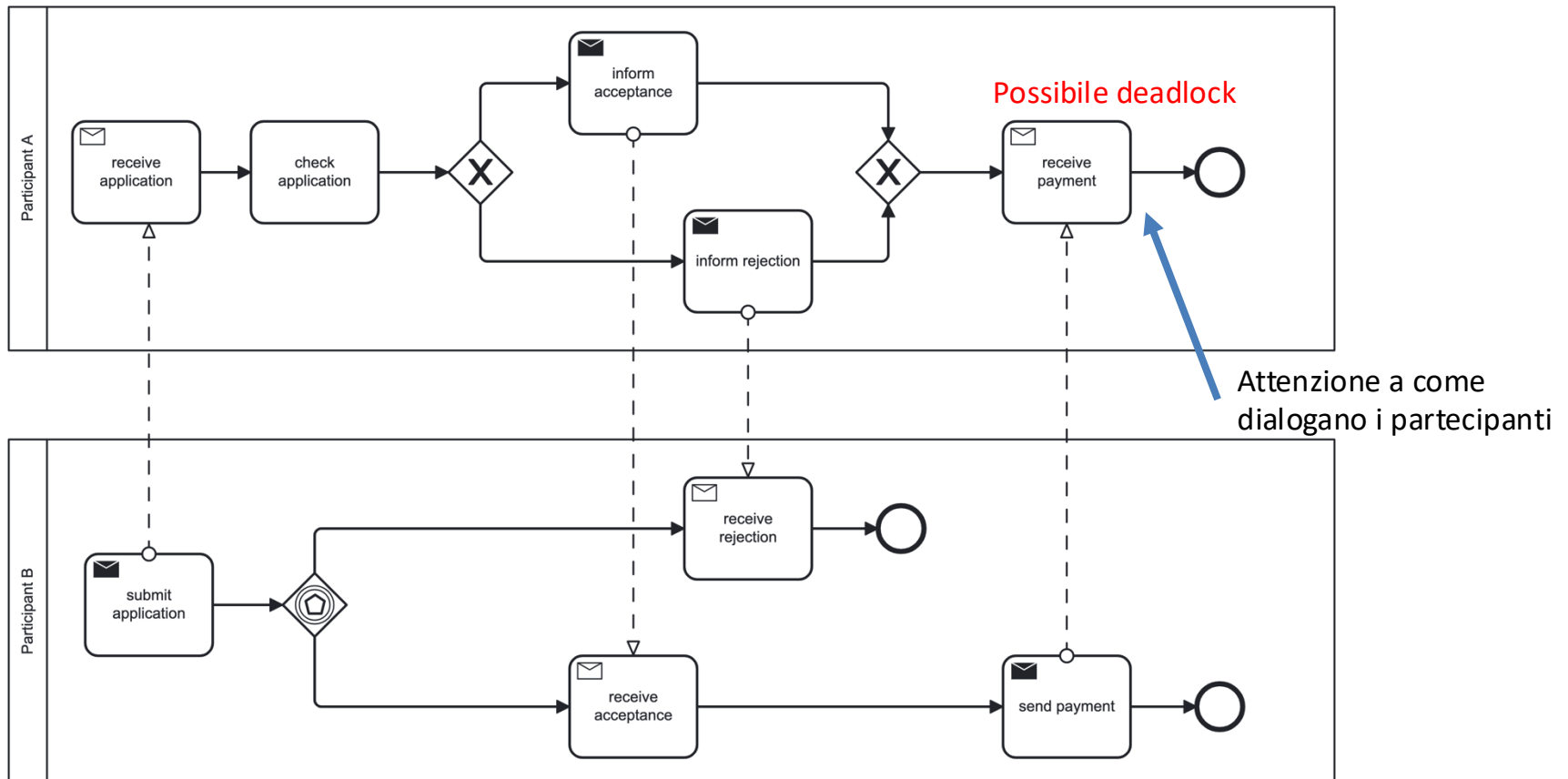


# Attenzione ai punti di decisione

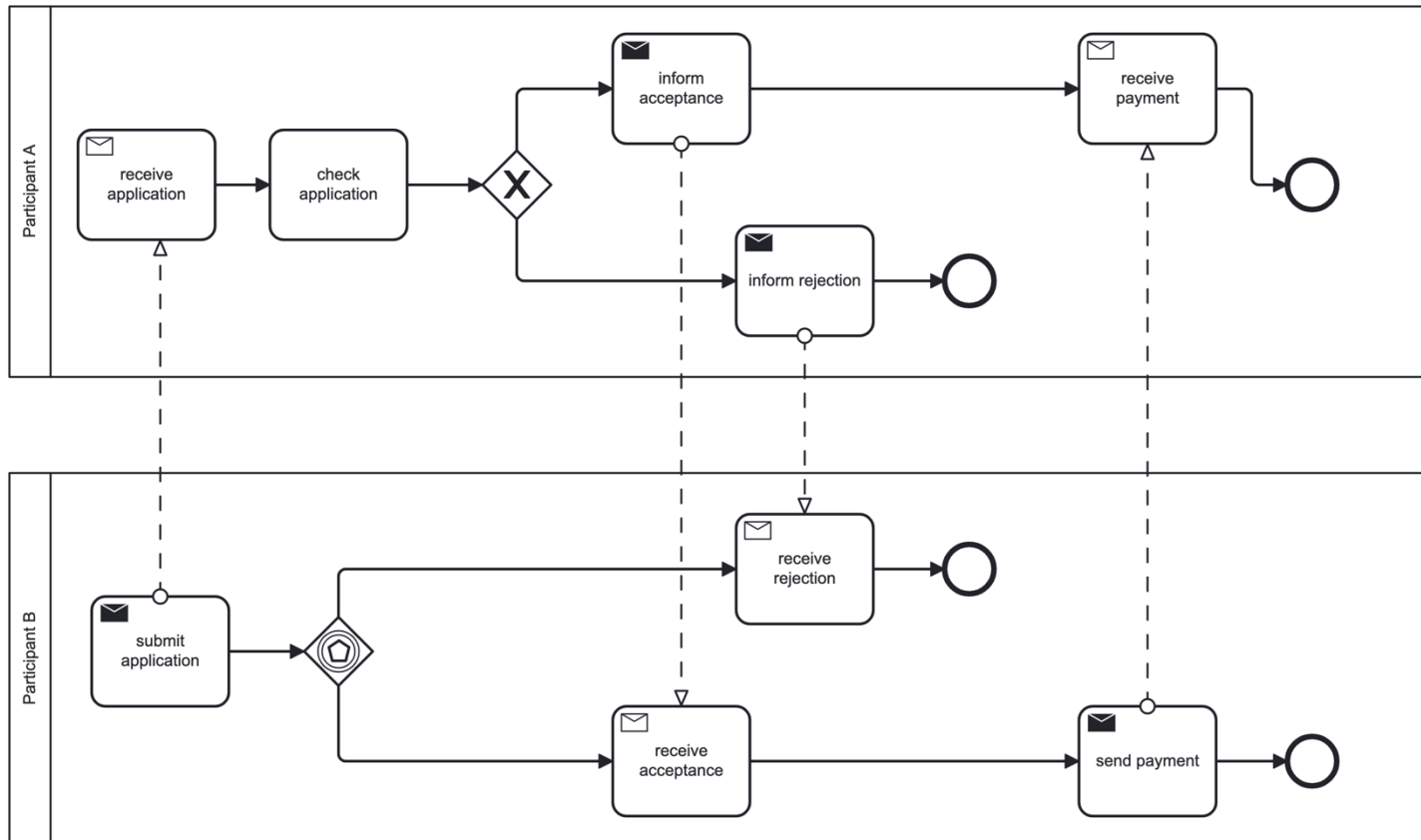
Event-based XOR  
(deferred choice)



# Attenzione ai flussi di esecuzione



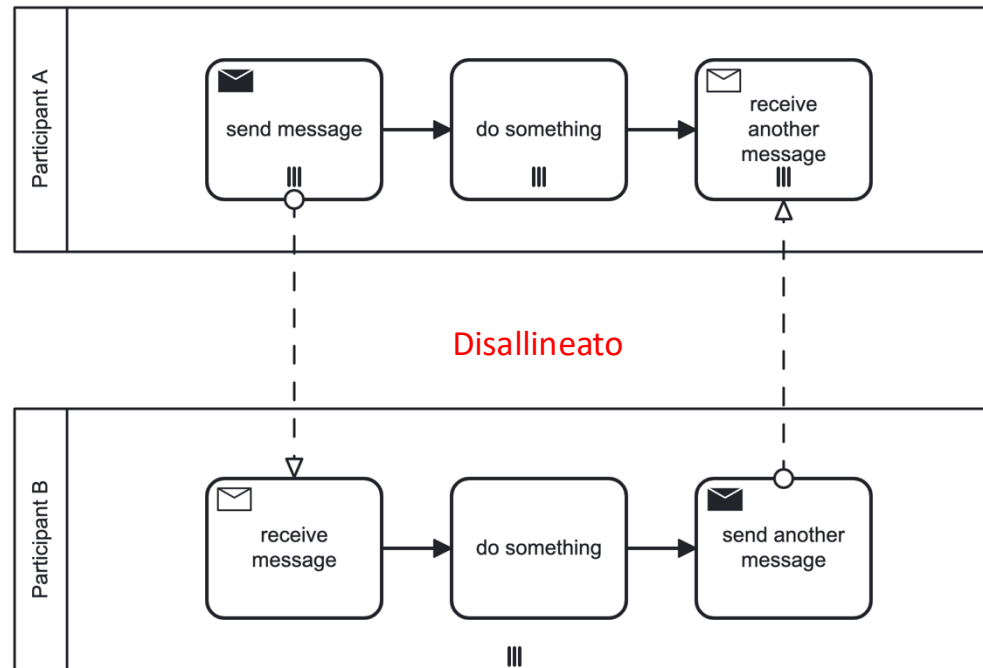
# Attenzione ai flussi di esecuzione



Versione  
corretta

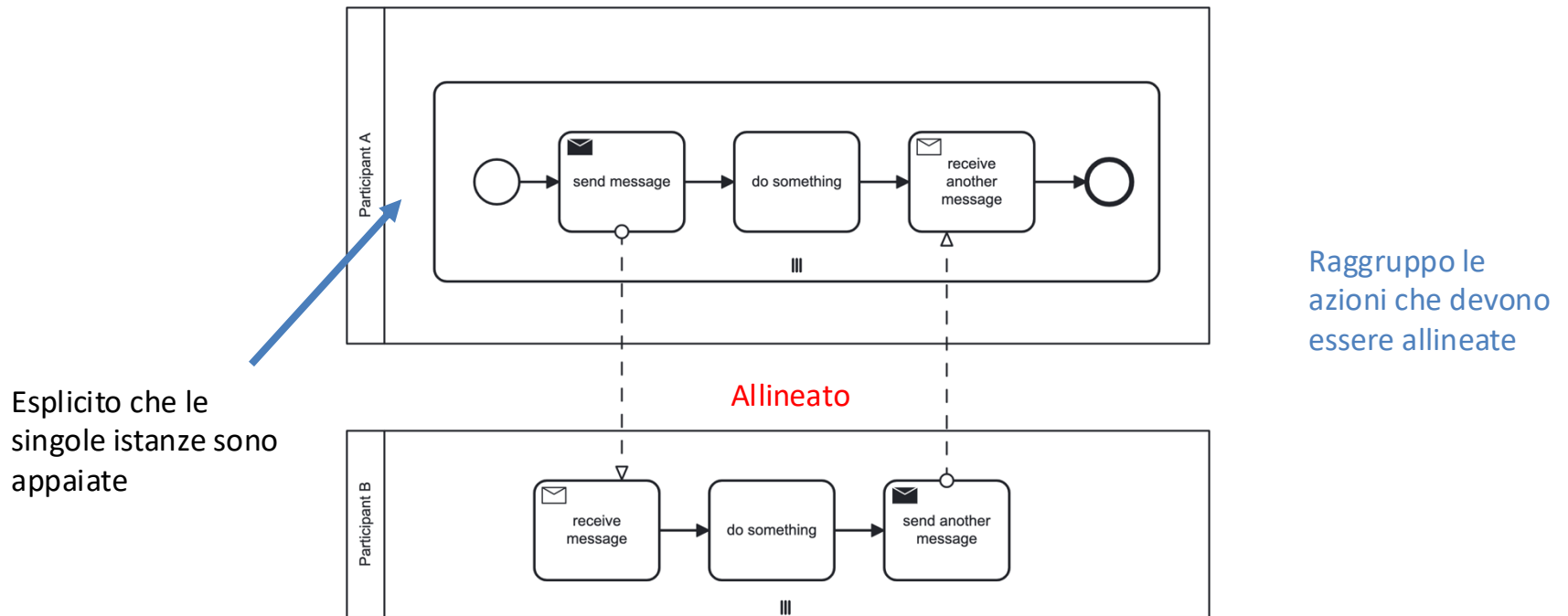


# Attenzione all'allineamento nelle istanze

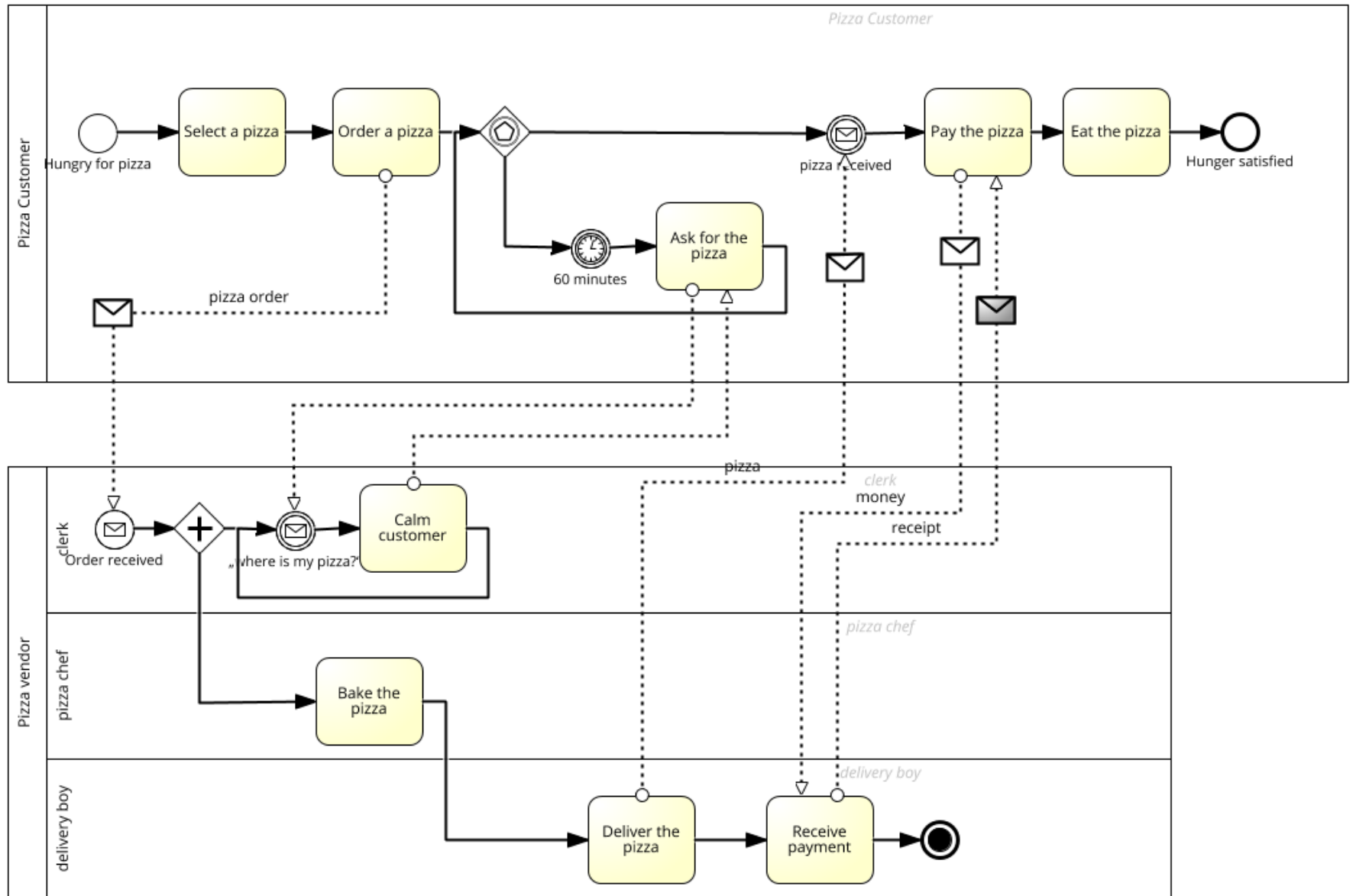


Come vengono  
allineate le singole  
richieste e  
risposte?

# Attenzione all'allineamento nelle istanze



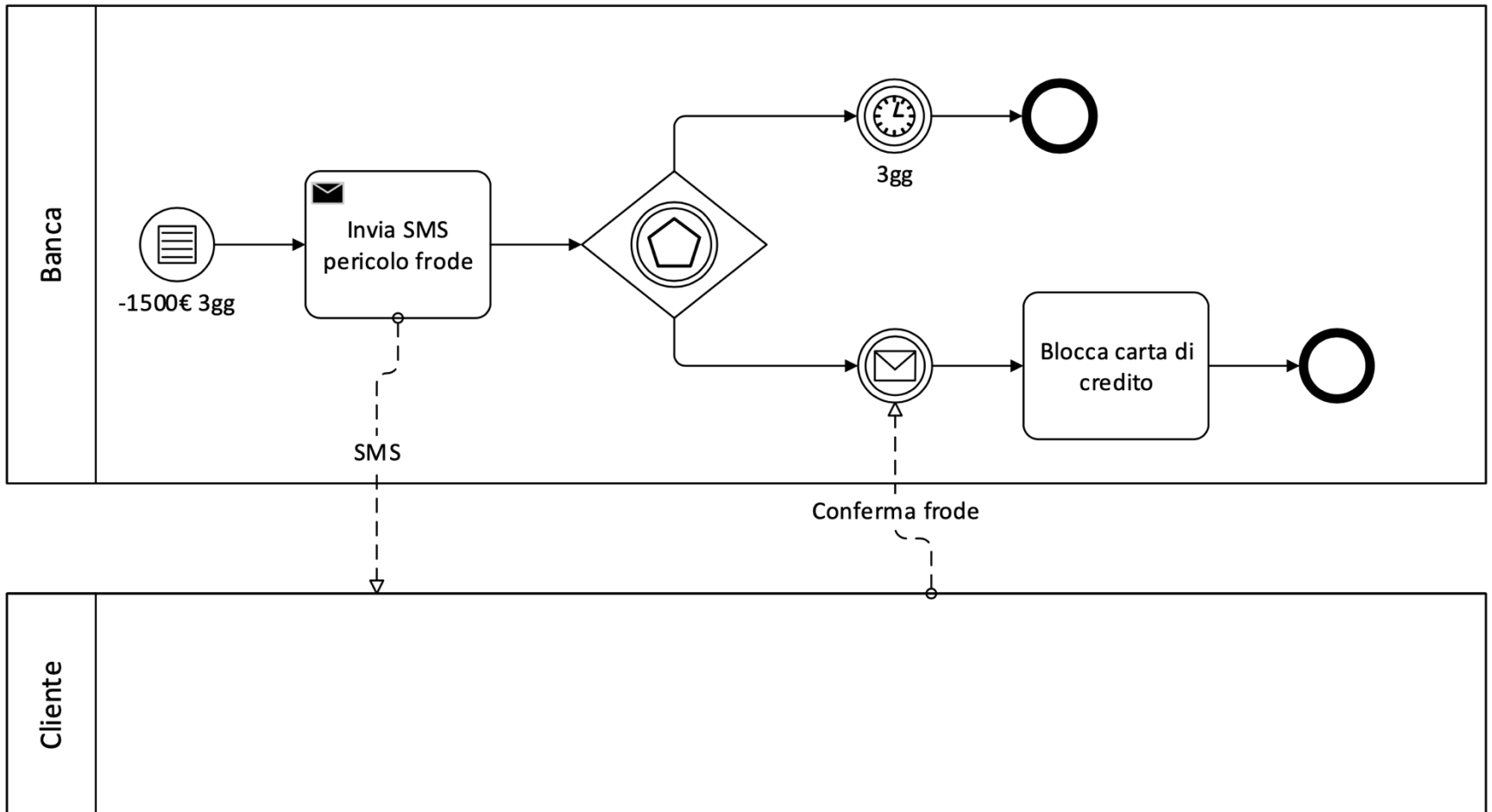
# Esempio completo



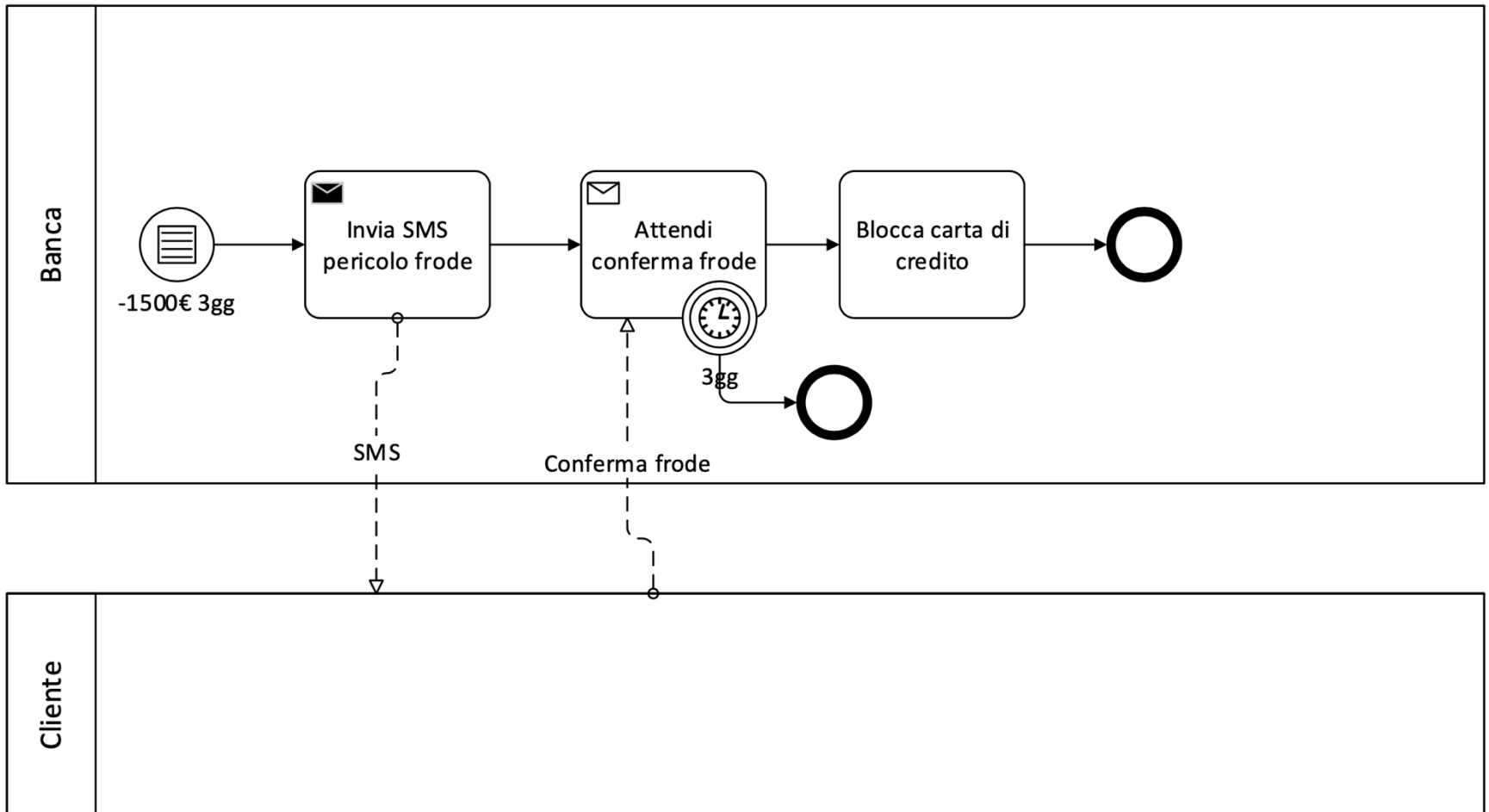
# Esercizio 1

«Il processo di identificazione frodi della banca FurboniBank è articolato come segue. Se i prelievi effettuati con la carta di credito di un cliente presso gli sportelli Bancomat superano la quota 1500€ a distanza di tre giorni, il sistema manda una notifica di tale sospetta attività via SMS al numero di cellulare del cliente, chiedendo di rispondere con il testo «FRODE» qualora non sia al corrente di tali movimenti. Se il sistema non riceve risposta entro tre giorni, il processo termina. Altrimenti la carta di credito del cliente viene bloccata.»

# Esercizio 1



# Esercizio 1

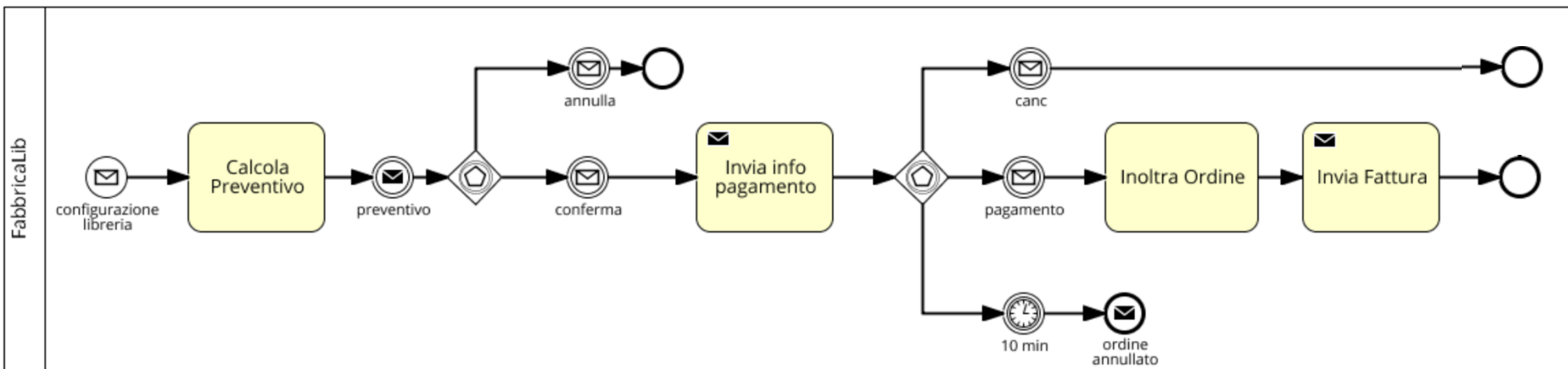


# Esercizio 2.1

La compagnia Alfomi offre ai suoi clienti un sistema per la creazione di librerie personalizzate.

Si modelli utilizzando BPMN il processo che descrive la vendita di una libreria. Il processo inizia quando la società riceve una richiesta di preventivo da parte del cliente. Sulla base delle informazioni fornite calcola il preventivo e lo invia al cliente. A questo punto rimane in attesa della decisione del cliente che può accettare il preventivo o rifiutarlo. Nel caso di annullamento il processo termina. Altrimenti, la società invia le informazioni di pagamento al cliente. La società attende quindi l'invio del pagamento che deve avvenire entro 10 minuti, allo scadere dei quali l'ordine viene annullato e l'annullamento viene notificato al cliente. Il cliente, in alternativa all'invio dei dati di pagamento, può inviare una notifica di annullamento dell'ordine che fa terminare il processo. Nel caso in cui il pagamento venga effettuato, la società inoltra l'ordine al proprio reparto di produzione e invia la fattura al cliente.

# Esercizio 2.1



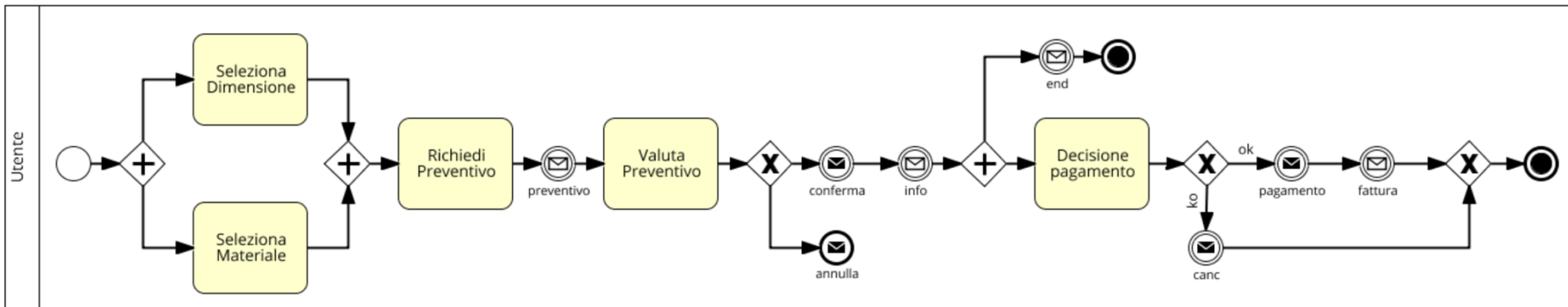


# Esercizio 2.2

La compagnia Alfomi offre ai suoi clienti un sistema per la creazione di librerie personalizzate. Si modelli utilizzando BPMN il processo che descrive l'acquisto di una libreria.

Il cliente che vuole acquistare una libreria deve innanzitutto sceglierne le dimensioni e scegliere il materiale di cui vuole che la libreria sia fatta. Queste due operazioni possono essere eseguite contemporaneamente. Una volta configurata la libreria, invia la richiesta di preventivo alla società Alfomi. Ricevuto il preventivo, il cliente può quindi scegliere se accettarlo e procedere all'acquisto o se annullare la procedura. Nel caso di annullamento il processo termina, previo invio di una notifica alla società. Altrimenti, il cliente conferma il suo ordine e attende la richiesta delle informazioni di pagamento da parte di Alfomi. Il cliente, ricevuti i dettagli di pagamento, può decidere se effettuare il pagamento o annullare l'acquisto e quindi cancellare l'ordine. Nel caso in cui il pagamento venga effettuato, rimane in attesa della fattura. Nel caso in cui decida di annullare, invia una notifica alla società. Dal momento in cui il cliente riceve le informazioni di pagamento fino a quando non invia una decisione, il cliente potrebbe ricevere una notifica di cancellazione da parte di Alfomi che porta all'interruzione del processo.

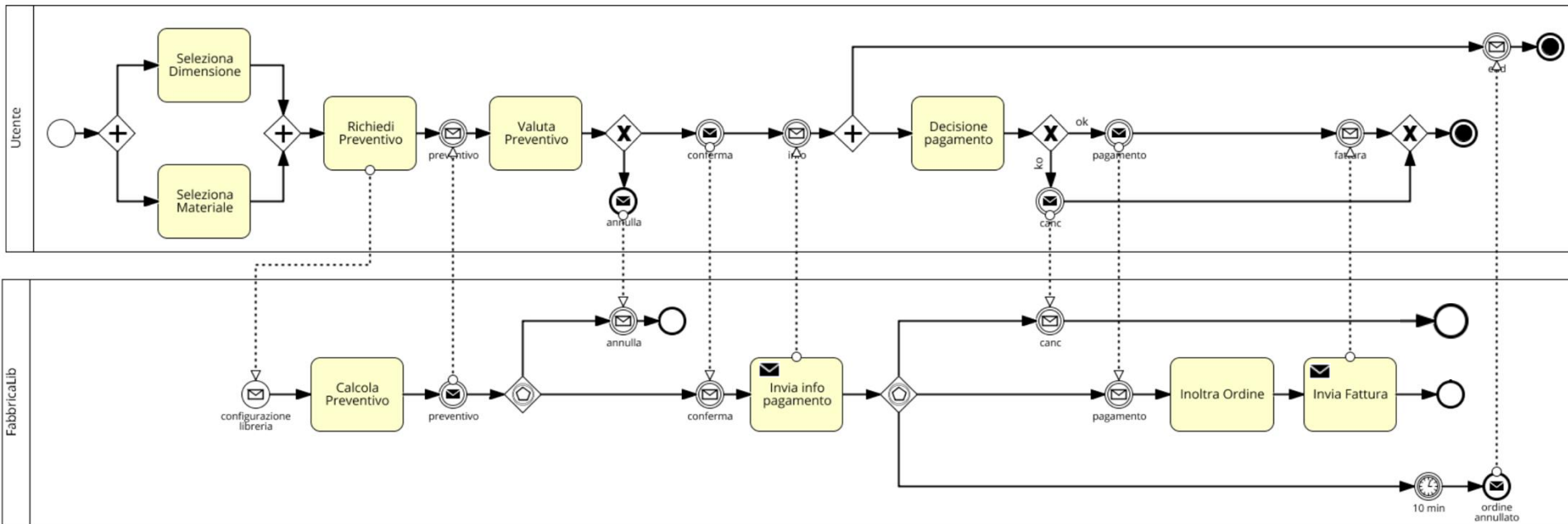
# Esercizio 2.2



# Esercizio 2.3

Partendo dai processi relativi ai punti 1 e 2, modellare la comunicazione tra processi mediante messaggi.

# Esercizio 2.3



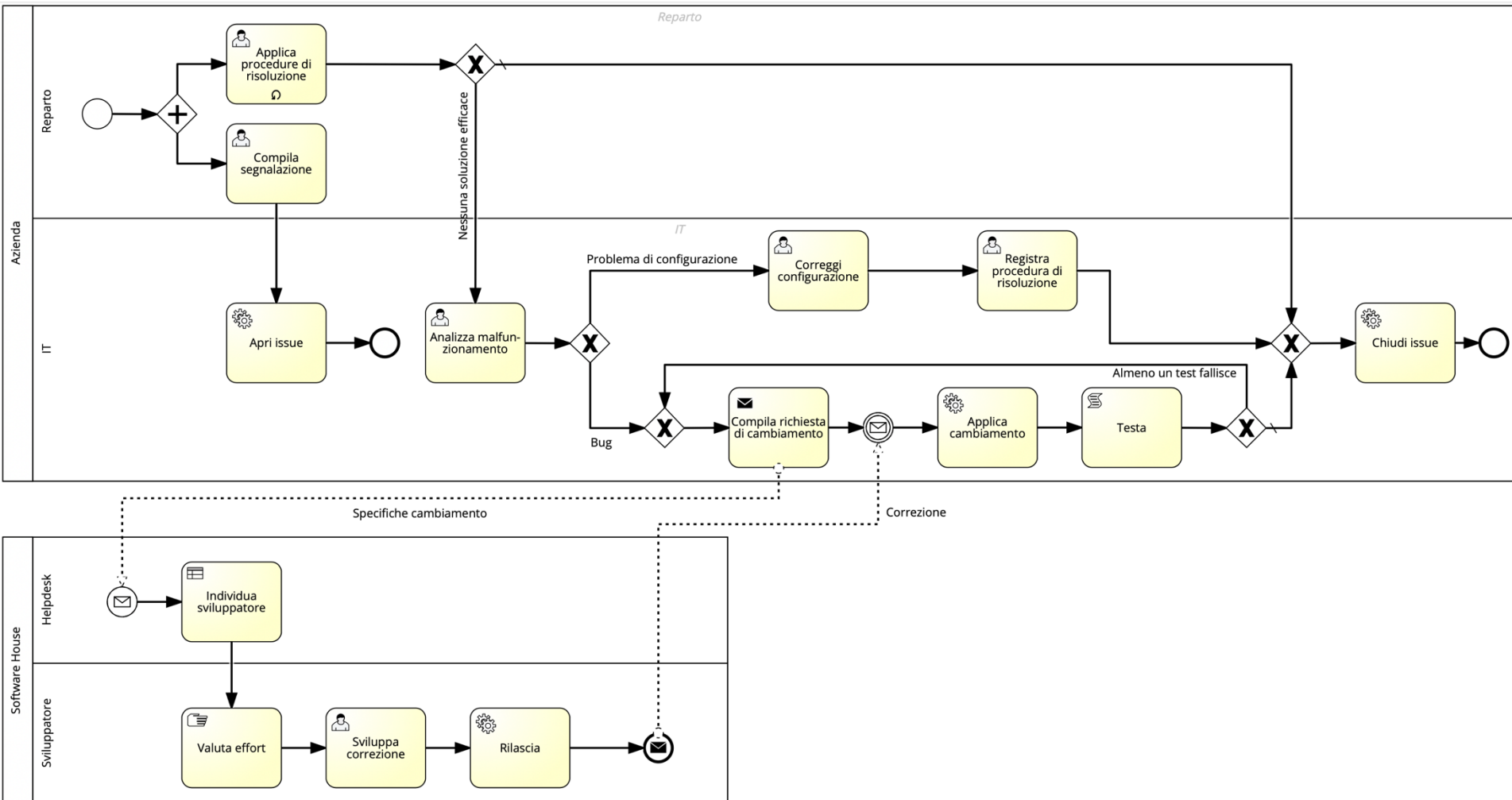
# Esercizio 3

Una azienda basa le sue operazioni su sistemi informatici. Quando un qualsiasi reparto riscontra un malfunzionamento a tali sistemi, deve aprire una segnalazione standardizzata al reparto IT, che aprirà un issue all'interno del sistema di tracciamento, e tentare di applicare le procedure di risoluzione note all'interno della knowledge base. Nel caso in cui una di queste procedure abbia successo, il reparto IT provvederà a chiudere l'issue; altrimenti, il reparto IT prende in carico la risoluzione tramite un'analisi del malfunzionamento. Se questa determina che il malfunzionamento è legato alla configurazione del sistema, il reparto IT provvede a correggere tale configurazione e registrare la procedura seguita all'interno della knowledge base per future necessità; altrimenti, trattandosi di un bug, il reparto IT invia una richiesta di cambiamento all'helpdesk della software house. Questo, ricevuta la richiesta, provvede ad individuare lo sviluppatore appropriato a svilupparla tramite le informazioni contenute nel sistema di gestione delle competenze. Una volta individuato, lo sviluppatore valuta l'effort necessario per la correzione, la sviluppa e la rilascia. Ricevuto il rilascio, viene applicato il cambiamento dal reparto IT dell'azienda che poi procede a dei test automatizzati. Se questi test falliscono, una nuova richiesta di cambiamento viene generata; altrimenti, l'issue viene chiusa.

# Esercizio 3

- Individuazione di pool e lane
- Evento di start: *riscontra un malfunzionamento a tali sistemi*: va bene evento generico, va bene errore, va bene condizione
- Chi applica le soluzioni della KB? Che tipo di multi-instance? Alternativa loop con XOR
- Alternative fine dello sviluppatore: attività con messaggio e poi end, oppure attività senza messaggio ed end con messaggio in throw

# Esercizio 3

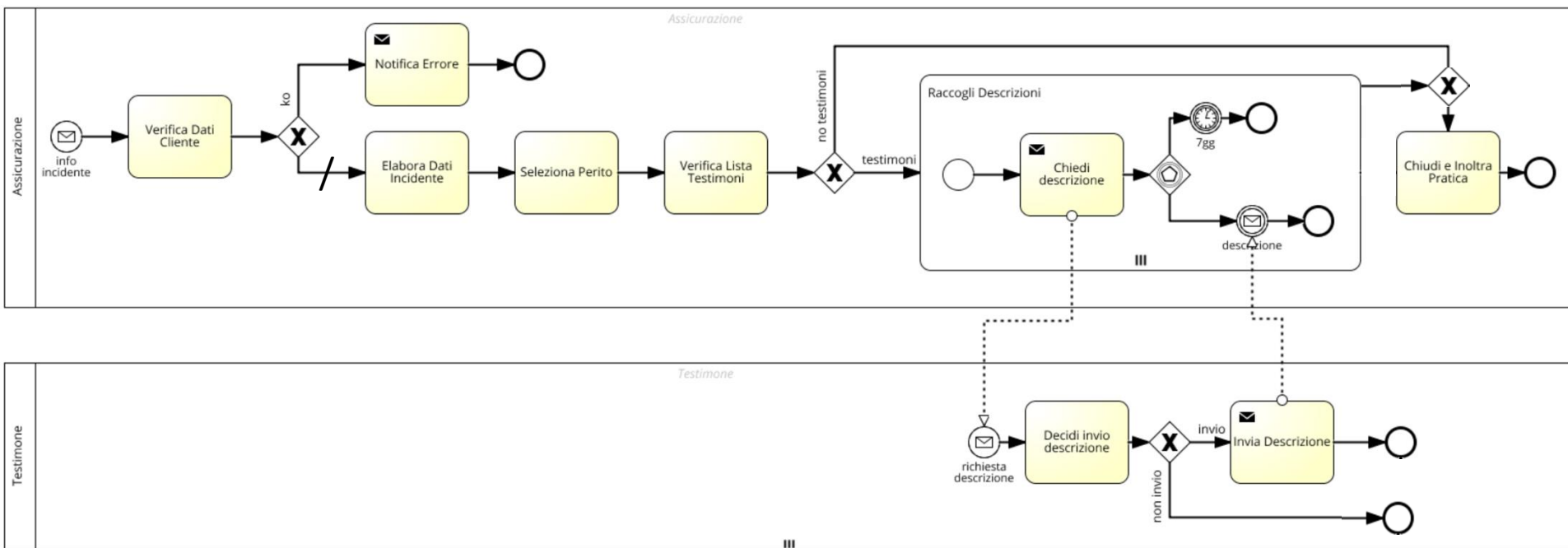


# Esercizio 4

Una compagnia di assicurazioni automobilistiche offre ai suoi clienti un sistema di inserimento di dati relativi a sinistri. Il processo inizia quando la compagnia di assicurazione riceve una notifica da parte di un cliente che contiene informazioni su un incidente stradale. Il sistema verifica i dati del cliente (nome, targa) e notifica al cliente eventuali errori. In questo caso il processo termina. In caso contrario, l'agenzia elabora i dati relativi all'incidente e seleziona un perito per la verifica dei danni. Infine, nel caso in cui il cliente abbia indicato il recapito di testimoni, la compagnia invia loro una richiesta per ottenere una descrizione dell'incidente. I testimoni possono non rispondere, quindi la compagnia attende 7 giorni, dopo i quali chiude la pratica e invia tutte le informazioni al perito. Modellare soltanto la compagnia di assicurazione e i testimoni.

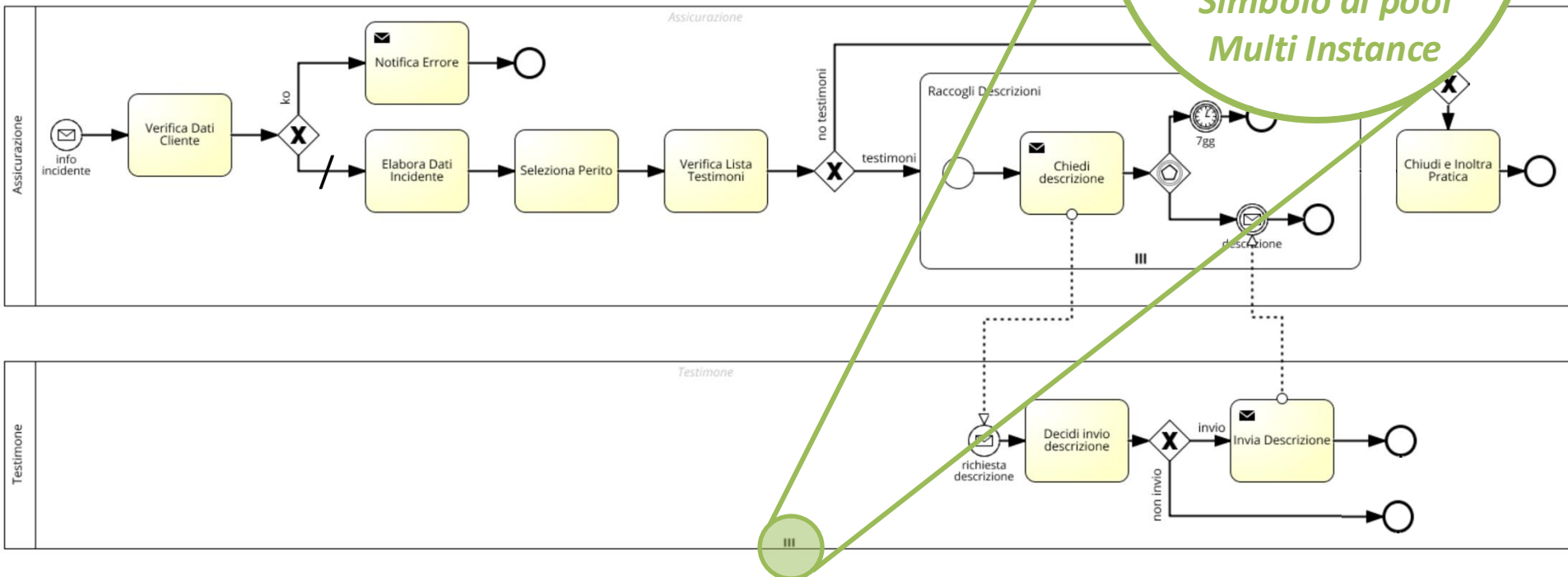


# Esercizio 4



# Esercizio 4

Una **Pool** è **Multi Instance** quando nella collaborazione fra i processi partecipano **più partecipanti**, i quali seguono **tutti lo stesso processo**.



# Esercizio 5

«La società MagiMug è specializzata nella realizzazione di tazze promozionali per conto di aziende, università, hotel, ecc. Per fare ciò, MagiMug acquista da un grossista tazze in ceramica bianca e si limita a serigrafare l'immagine richiesta dal cliente.

Quando il numero di tazze in magazzino è inferiore alle 10000 unità, la divisione amministrativa di MagiMug prepara un ordine di acquisto e lo invia al proprio grossista. Una volta ricevuto l'ordine, il grossista provvede a preparare un container da spedire a MagiMug. Se durante tale attività il contenuto del container viene fatto cadere e si danneggia, l'attività viene interrotta, viene stimata l'entità del danno e, simultaneamente, rimosso il materiale danneggiato dal container, quindi l'attività di preparazione del container riprende. Non appena il container è pronto, viene inviata la richiesta di trasporto ad un autotrasportatore, il quale carica il container su un autotreno e lo consegna a MagiMug. Una volta ricevuto il container, i magazzinieri di MagiMug provvedono a controllare l'integrità del contenuto: se è tutto a posto, ciò viene segnalato all'autotrasportatore, il quale notifica a propria volta il successo dell'attività al grossista, ed il processo termina. Se invece parte della merce risulta danneggiata, l'autotrasportatore riporta il container dal grossista ed il processo termina.»

# Esercizio 5

