

MODELLEREN VAN PROCESSEN

3/3

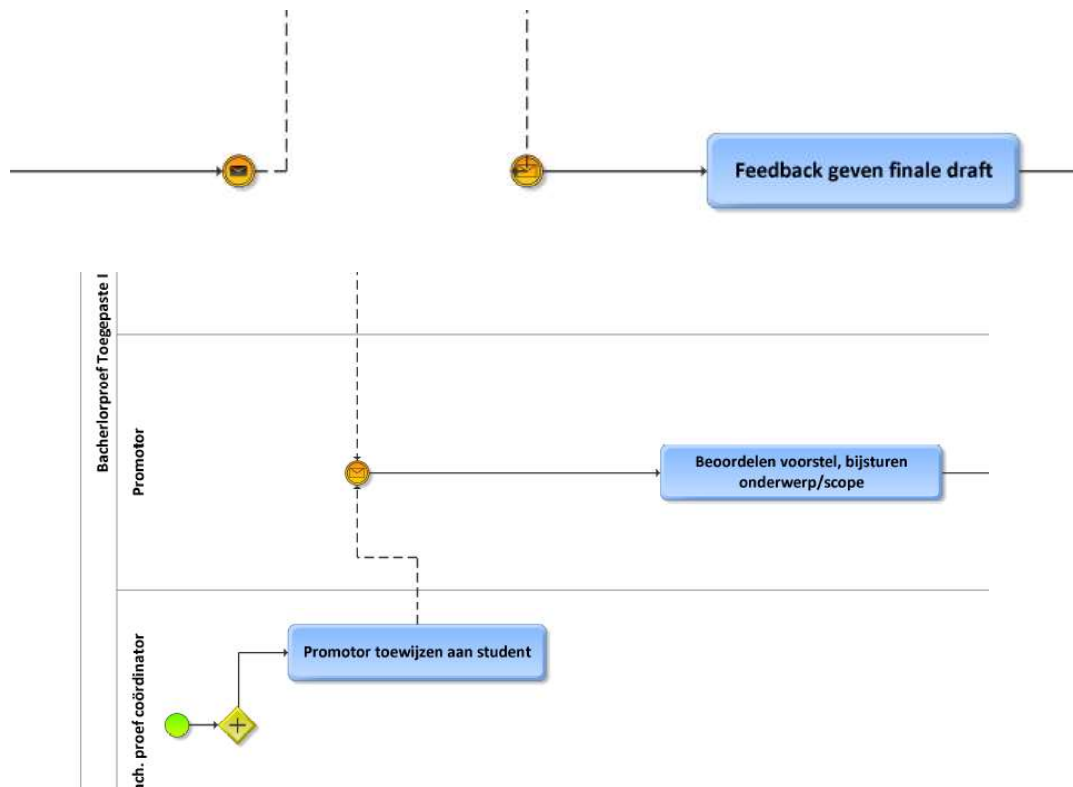
17-4-2023

1

VAAK VOORKOMENDE FOUTEN

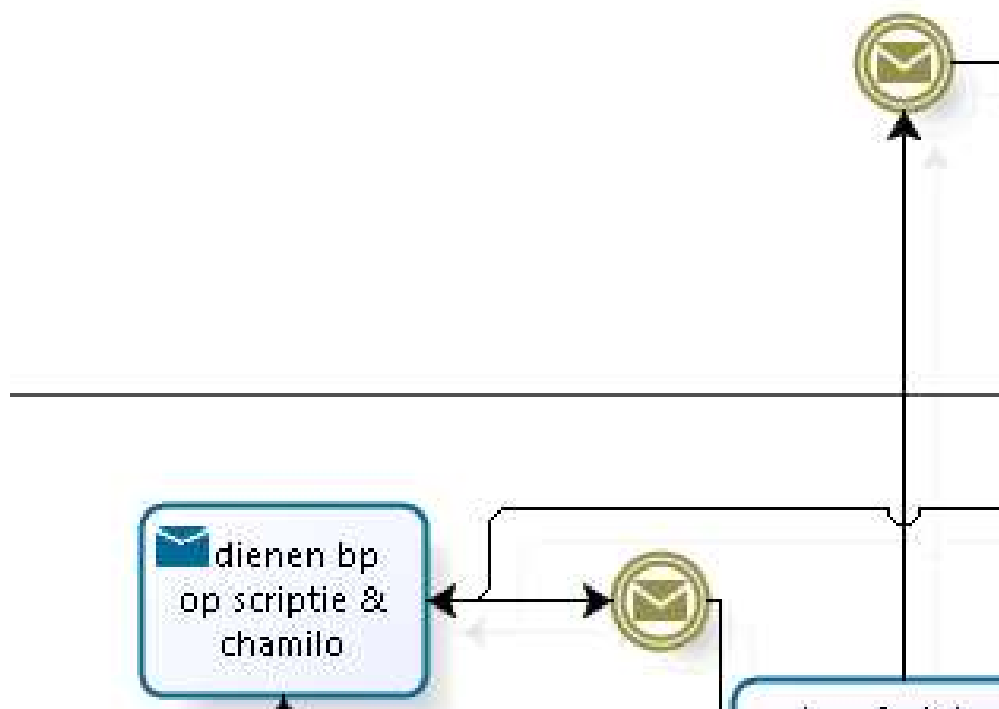
17-4-2023

2



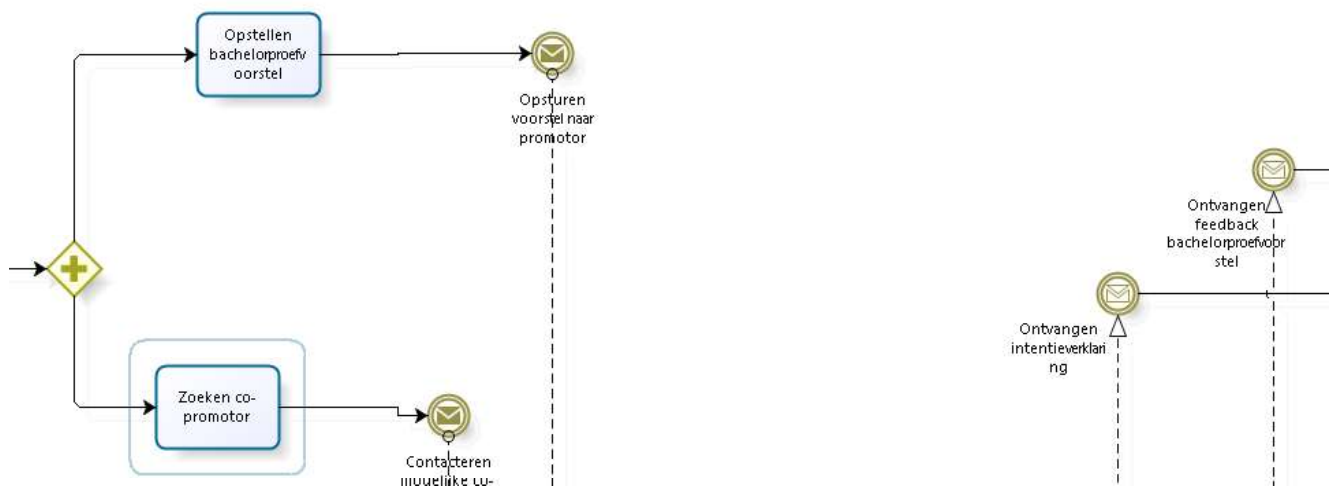
17-4-2023

3



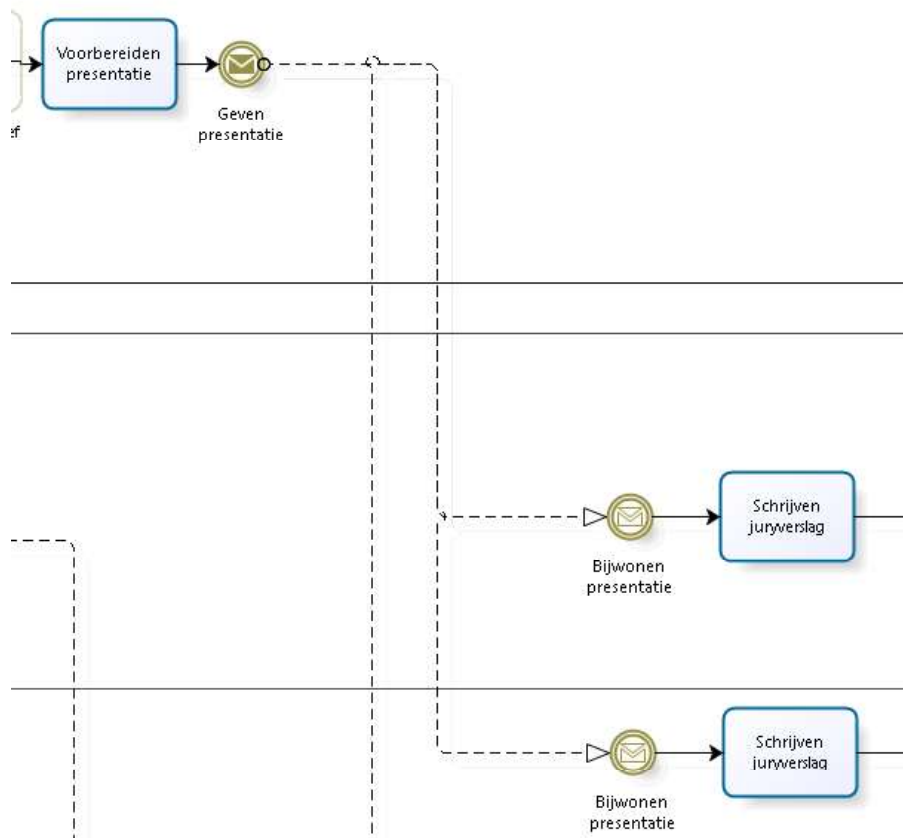
17-4-2023

6



17-4-2023

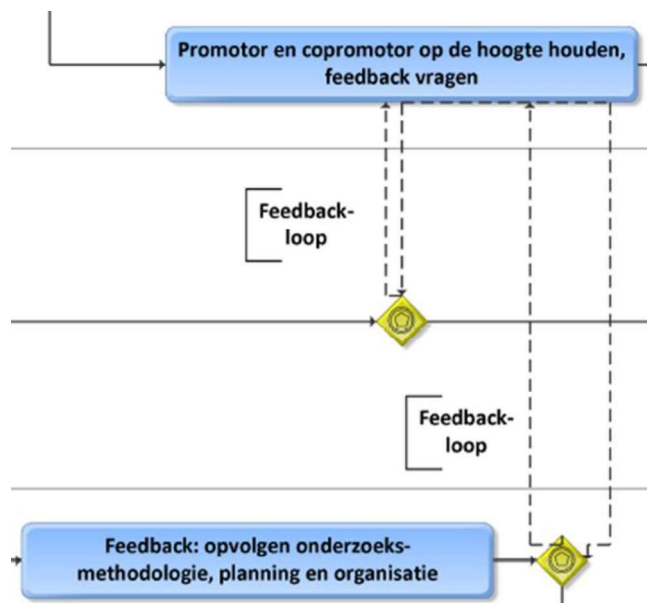
7



17-4-2023

8

Zoek de fout



17-4-2023

9

Events

	Top-Level	Event Sub-Process Interrupting	Event Sub-Process Non-Interrupting	Catching	Boundary Interrupting	Boundary Non-Interrupting	Throwing	End
None: ongetypeerde gebeurtenis, zoals een startpunt of statusverandering.								
Message: verzenden en ontvangen van berichten.								
Timer: cyclische timer gebeurtenissen, punten in de tijd, tijdspanne of time-out.								
Escalation: escaleren naar een hoger niveau van verantwoordelijkheid.								
Conditional: reactie op veranderende procescondities of integratie van BusinessRules.								
Link: twee corresponderende link events zijn gelijk aan een 'Sequence Flow'.								
Error: opvangen of werpen van gedefinieerde fouten.								
Cancel: reactie op geannuleerde transacties of de start van een annulering.								
Compensation: omgaan of starten van een compensatie.								
Signal: signalering tussen processen. Kan meerdere keren worden ontvangen.								
Multiple: ontvang één van een set gebeurtenissen, start alle gedefinieerde gebeurtenissen.								
Parallel Multiple: ontvang alle uit een set parallelle gebeurtenissen.								
Terminate: zet de onmiddellijke beëindiging van een proces in gang.								

17

11

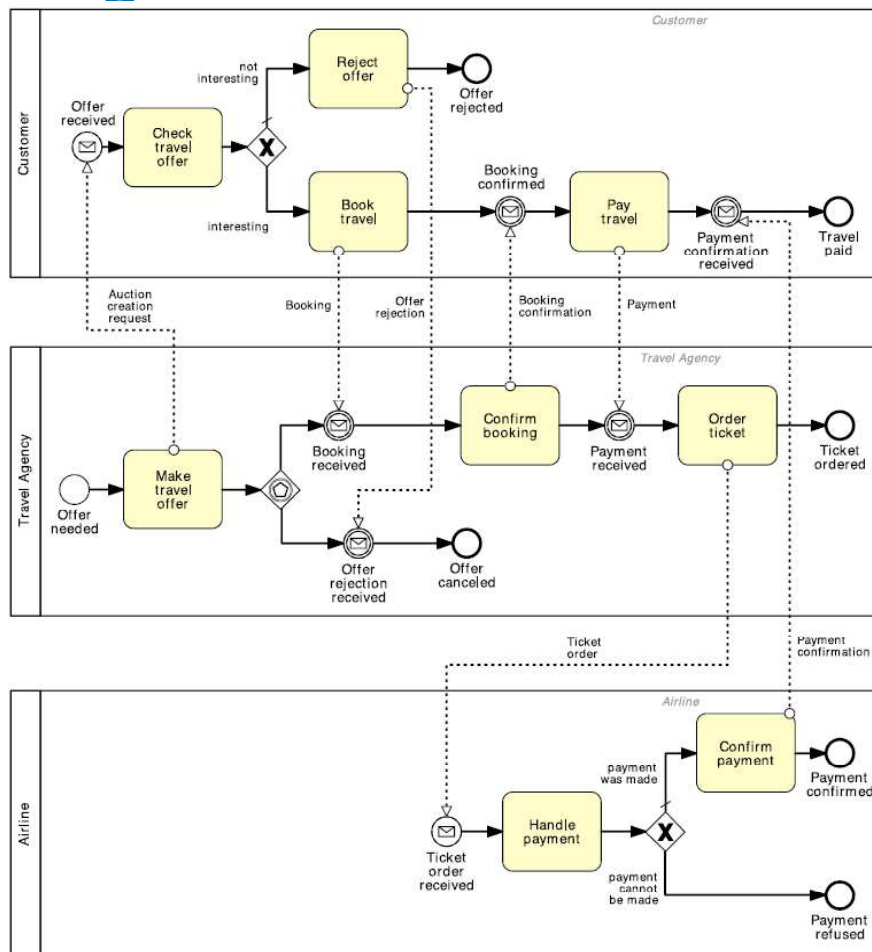
Wat kan beter?



17-4-2023

12

Herhaling: zoek de fout



13

Inhoud

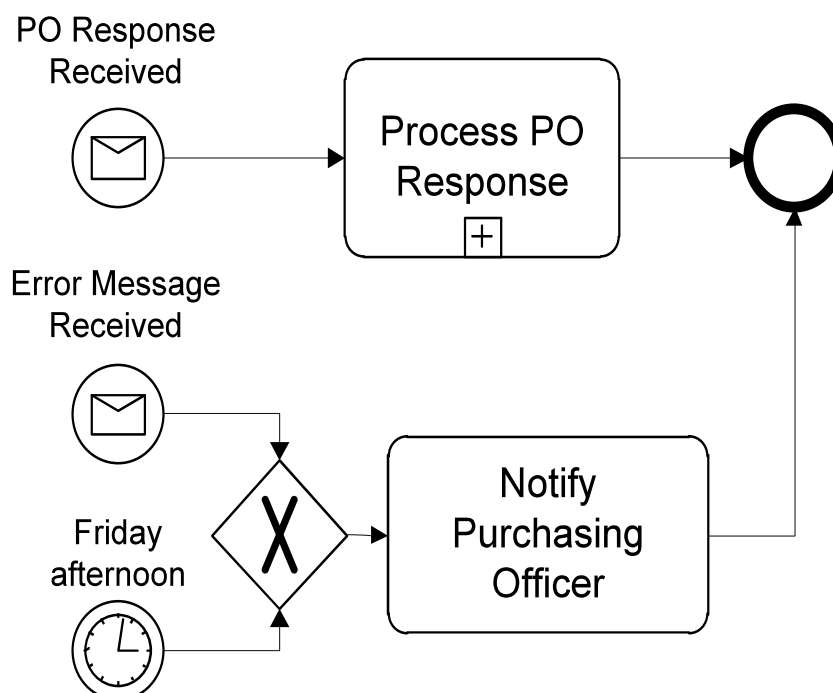
- (speciale) types van subprocessen
- loops
- exceptions
- compensaties

17-4-2023

15

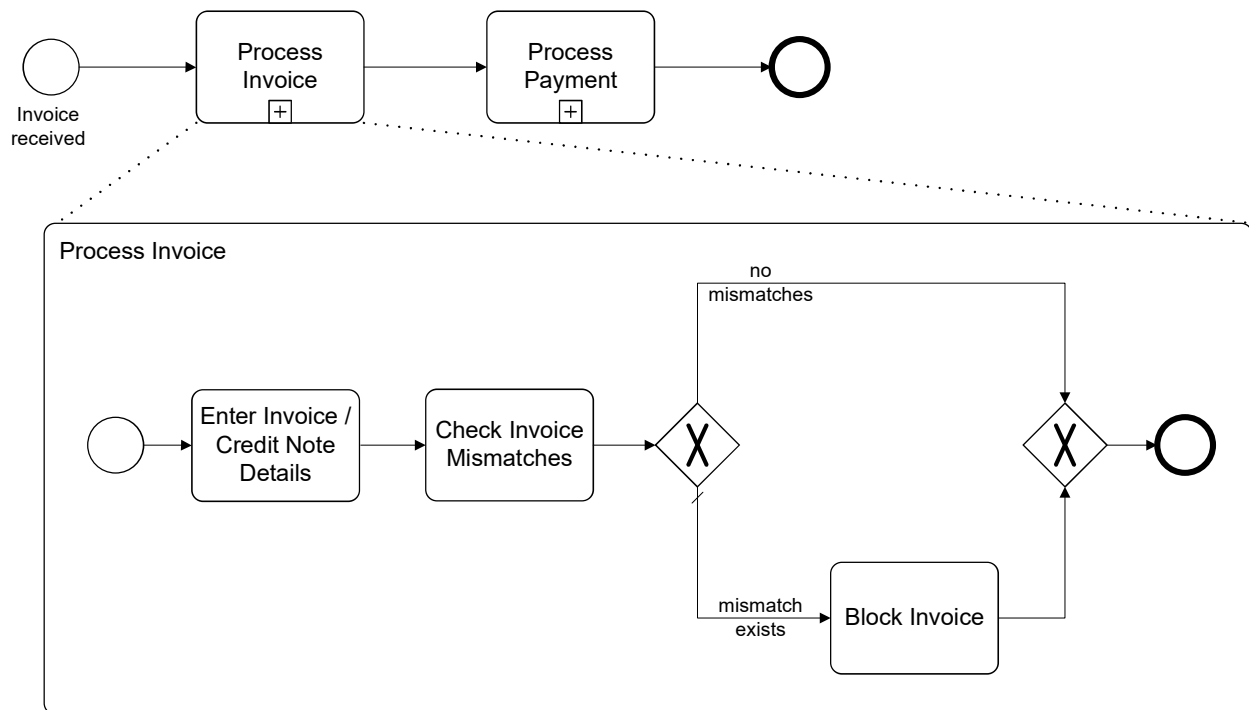
Interludium: multiple start events

Het eerste event dat zich voordoet zal het proces triggeren.



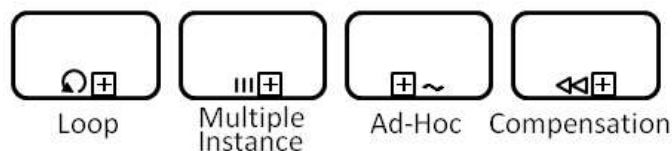
16

Subprocesses: herhaling



Subprocessen: speciale types

- De speciale types samengevouwen subprocessen zijn:



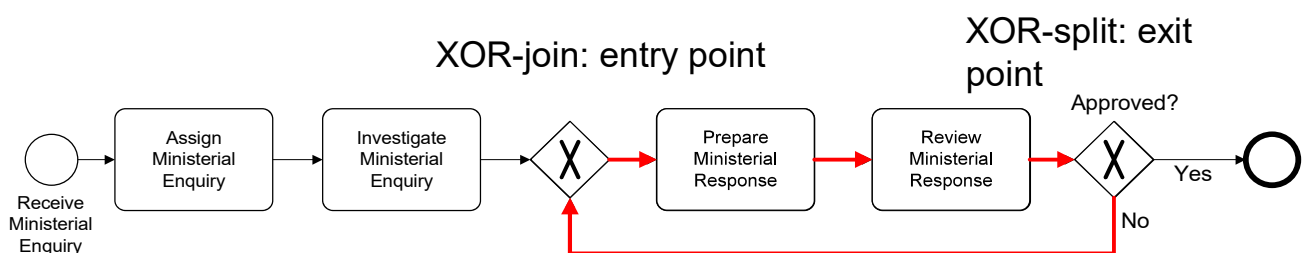
- Een *lus* (loop)-subproces
- Een *meerdere verzoeken* (multi-instance)-subproces
- Een *ad-hoc* proces: groep van activiteiten die geen voorgedefinieerde sequentiële relaties hebben.
- Een *compensatie*-subproces: groep van activiteiten die uitgevoerd worden bij het 'terugdraaien' van een transactie om activiteiten te compenseren die uitgevoerd werden tijdens de normale stroom van een proces

Loops: willekeurige cyclussen

- Een **cyclus** is een flow die teruggaat naar een vroeger punt in het proces.
- Wordt gebruikt om bepaalde onderdelen van het process te herhalen.

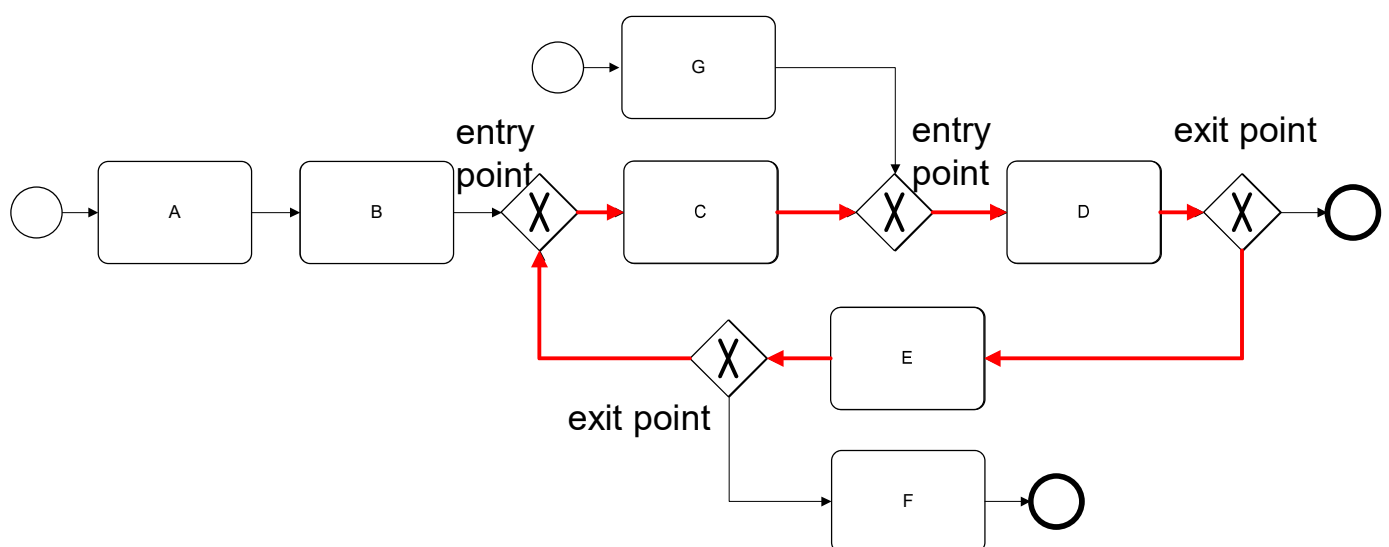
Voorbeeld: verwerking van vragen aan minister

Een antwoord op een ministeriële vraag wordt voorbereid door een kabinetsmedewerker en nagekeken door de kabinetschef, die het kan goedkeuren of afkeuren.



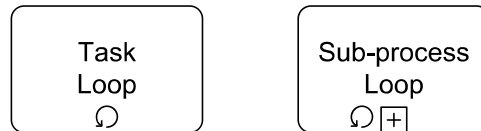
Willekeurige lussen (vervolg)

- Willekeurig = ongestructureerd, d.w.z. er kunnen meerdere ingangen en uitgangen zijn.



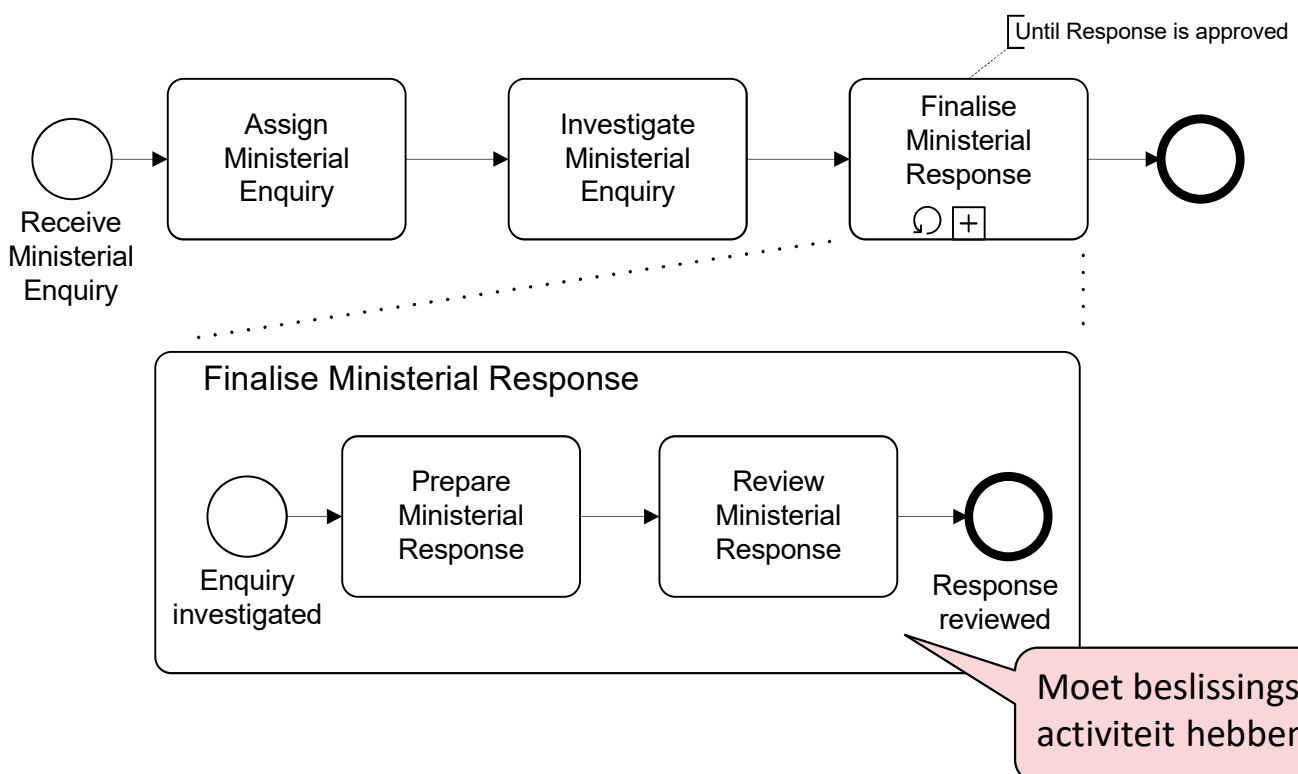
Gestructureerde herhaling: activity loop

- BPMN voorziet in een *activity loop*, die toelaat om een taak of subproces te herhalen



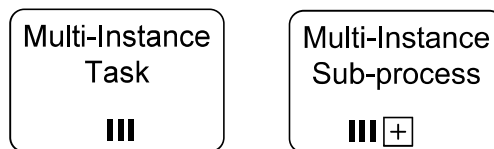
- Het belangrijkste verschil is dat de activity loop gestructureerd is, en dus maar één in- en uitgang heeft.
- Tip: gebruik de activity loop als de herhaling gestructureerd is.
- Probleem: gebruik subproces moeilijk indien meerdere zwembaden.

Voorbeeld: gestructureerde herhaling



Parallele herhaling: Multi-Instance activity

- De multi-instance activity voorziet in een mechanisme om aan te geven dat een activiteit of subprocess **meerdere keren gelijktijdig** wordt uitgevoerd.

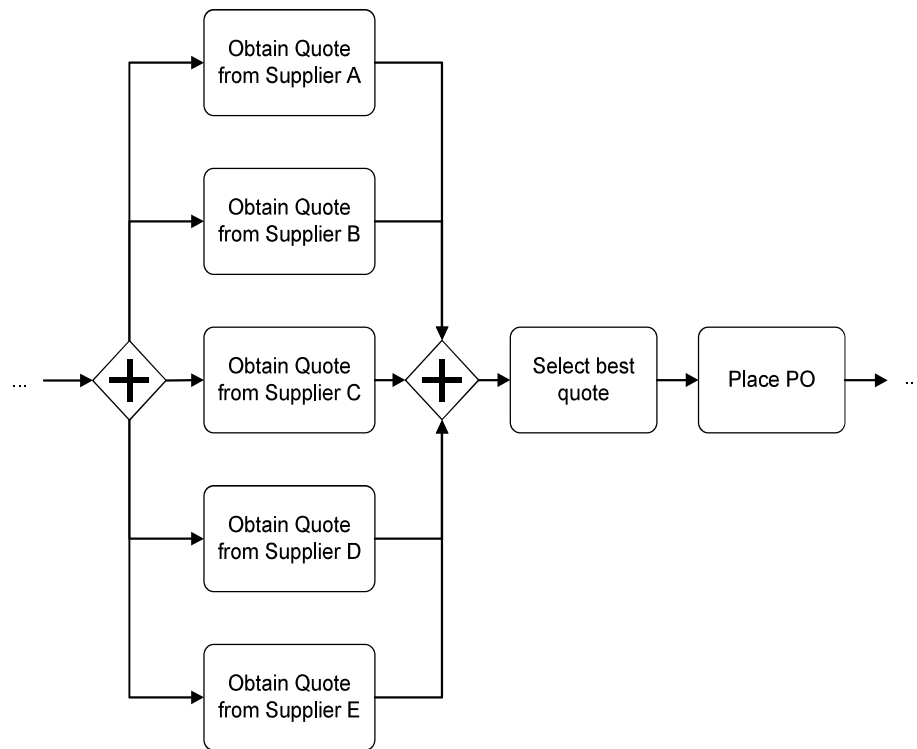


- Nuttig als dezelfde activiteit moet uitgevoerd worden voor verschillende entiteiten of data-items zoals
 - Aanvragen van een offerte bij verschillende leveranciers
 - Checken van de beschikbaarheid van alle artikelen in een bestelling
 - Verstuur en verwerk vragenlijsten van verschillende getuigen bij een ongeval voor het beoordelen van de verzekeringsclaim

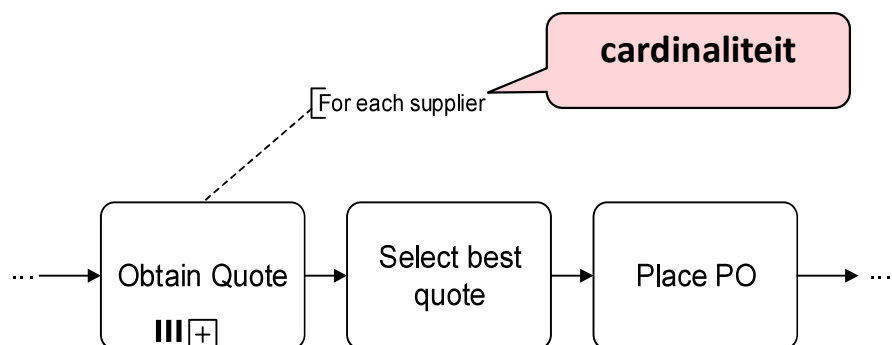
Voorbeeld: multi-instance activity

- In aankoopprocessen wordt typisch een offerte gevraagd aan verschillende "preferred suppliers" (veronderstelling: er zijn er 5)
- Nadat alle offertes binnen zijn, worden ze geëvalueerd en de beste wordt geselecteerd. Een aankooporder wordt geplaatst.

Oplossing zonder multi-instance activity



Oplossing met multi-instance activity

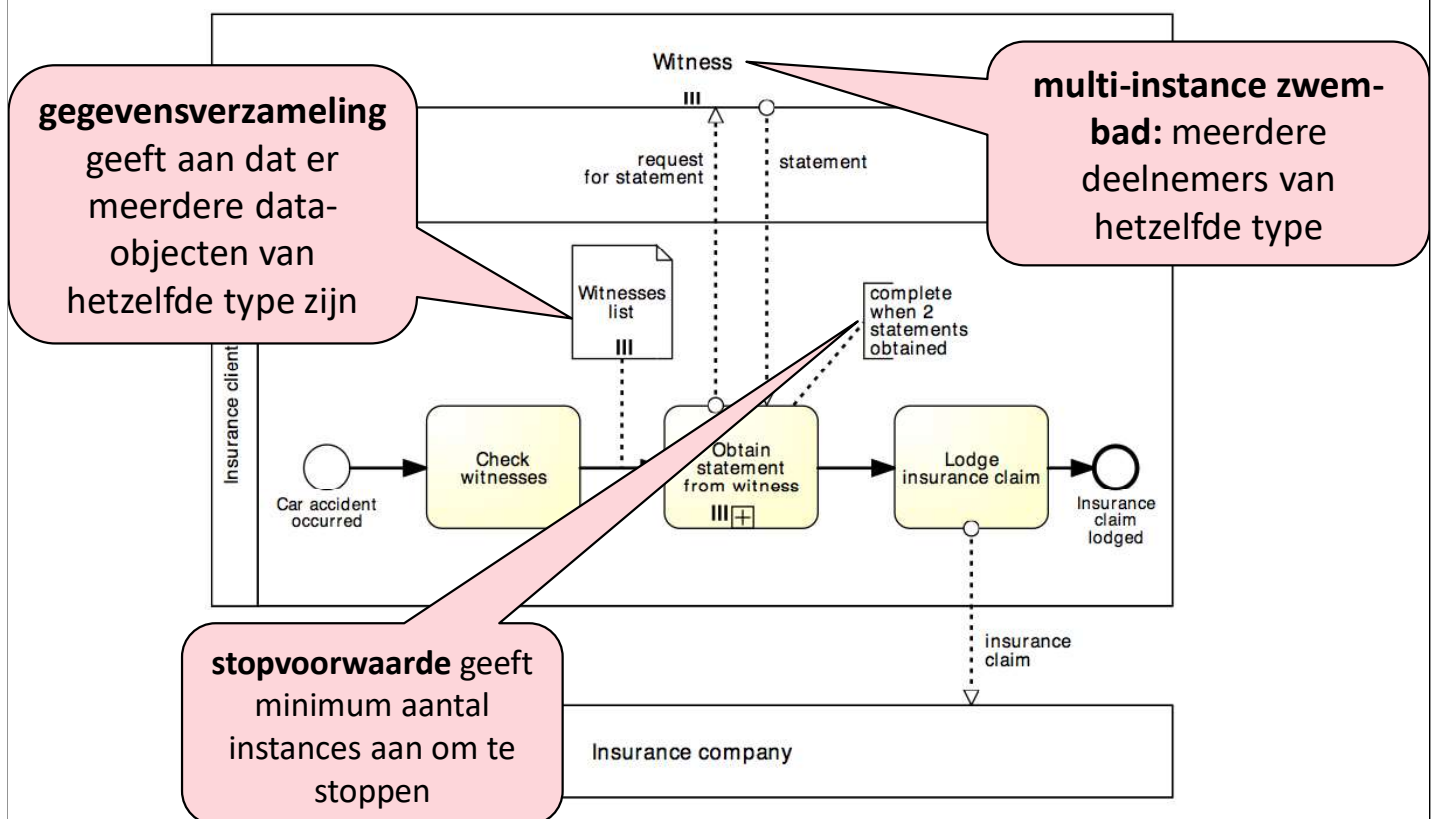


Voorbeeld: multi-instance activity

Na een auto-ongeval wordt door een verzekeringsmakelaar een verklaring gevraagd van de aanwezige getuigen, om de verantwoordelijke(n) te kunnen bepalen. Van zodra twee verklaringen binnengekomen zijn, wordt de claim doorgestuurd naar de verzekeringsmaatschappij, zonder dat nog moet gewacht worden op de andere verklaringen.

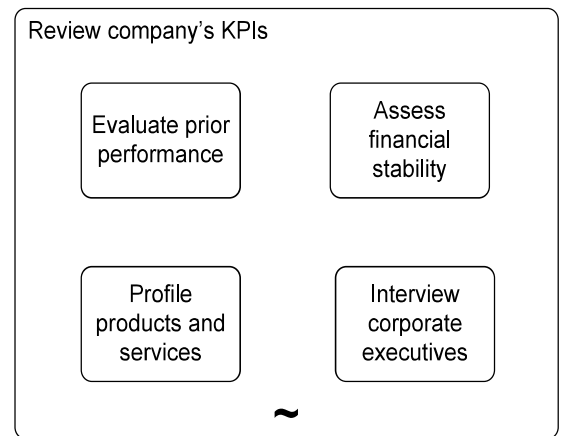
PS: iedereen overleefde het ongeval

Oplossing met multi-instance activity



Ongecontroleerde herhaling: Ad-hoc sub-process

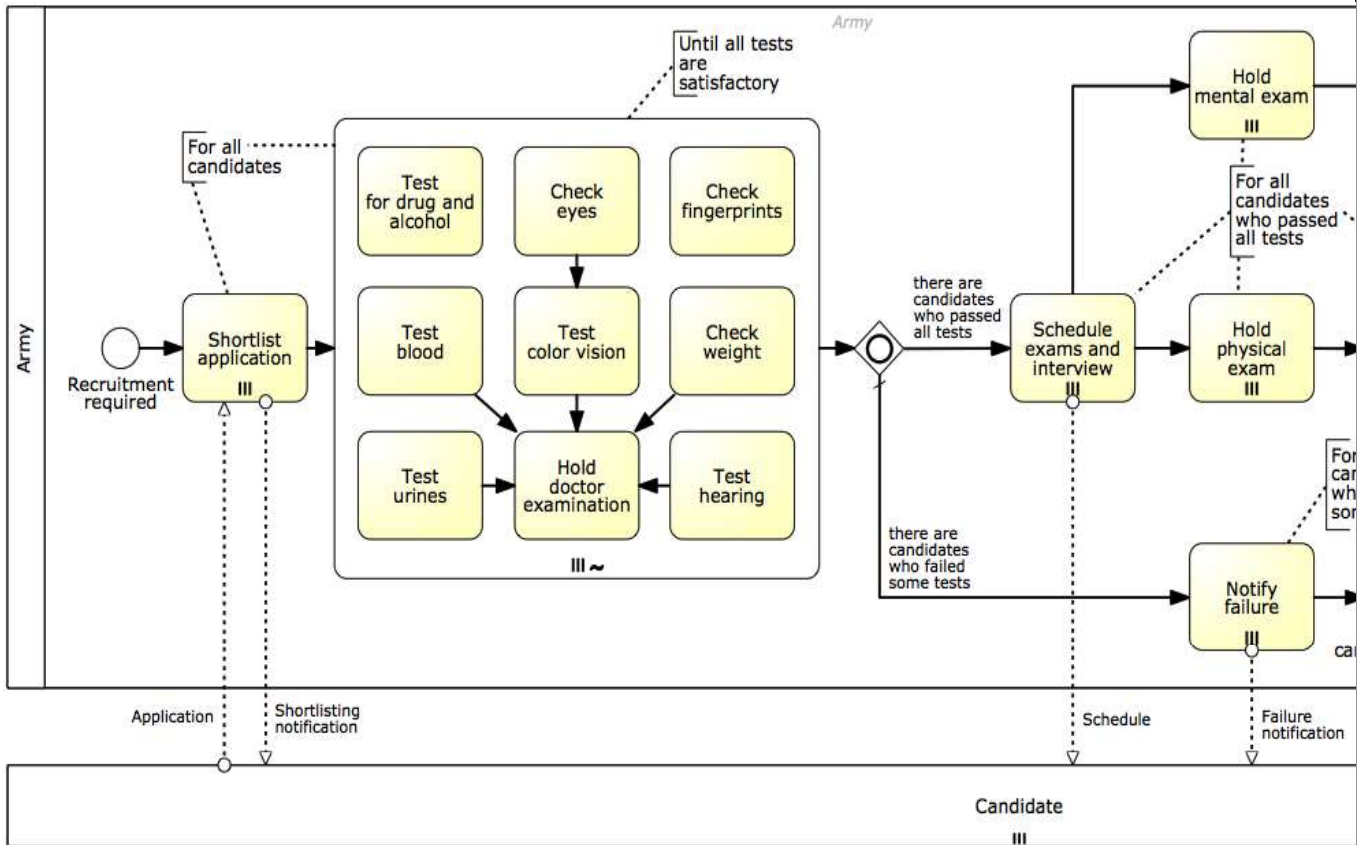
- Het ad-hoc subprocess bevat activiteiten (taken en subprocessen) die in willekeurige volgorde en op een willekeurig tijdstip mogen uitgevoerd worden.
- Indien voor bepaalde activiteiten volgorde toch belangrijk is, dan aangeven met pijlen
- Kan (ook) gebruikt worden in een vroege versie van het processchema, als de volgorde nog niet gekend is.
- Aangegeven door een tilde (~)



Voorbeeld: ad-hoc subprocess

Een selectieprocedure voor het leger start met een shortlist van alle kandidaten op basis van de sollicitaties. Deze kandidaten worden dan opgebeld om volgende testen te ondergaan: drugs en alcohol, ogen, kleurenzicht, gehoor, bloed, urine, gewicht, vingerafdrukken en een doktersonderzoek. Het kleurenzicht kan pas getest worden na de oogtest en de dokter kan zijn onderzoek pas starten nadat kleurenzicht, gehoor, bloed, urine en gewicht getest zijn. Bovendien kan het voor sommige kandidaten nodig zijn om bepaalde testen te herhalen om correcte resultaten te krijgen. Zo kan het zijn dat de bloedtest moet herhaald worden als de kandidaat teveel suiker gegeten heeft in de laatste 24u. Kandidaten die slagen op alle testen, moeten tenslotte een fysieke en een mentale test afleggen, gevolgd door een interview. Alleen zij die ook goed scoren op deze twee testen en goed presteren tijdens het interview, worden aangenomen in het leger.

Oplossing: ad-hoc subprocess



Oefening: leningsaanvraag

Als een leningsaanvraag binnenkomt bij de bank, wordt eerst de aanvraag op volledigheid nagekeken. Als die onvolledig blijkt, wordt ze teruggestuurd naar de aanvrager, die dan de ontbrekende informatie kan invullen en de aanvraag terugsturen naar de bank. Dit proces wordt herhaald tot de aanvraag volledig is ingevuld.

Modelleer dit proces als een gestructureerde en ongestructureerde loop. Modelleer ook de klant.

Exceptions

17-4-2023

35

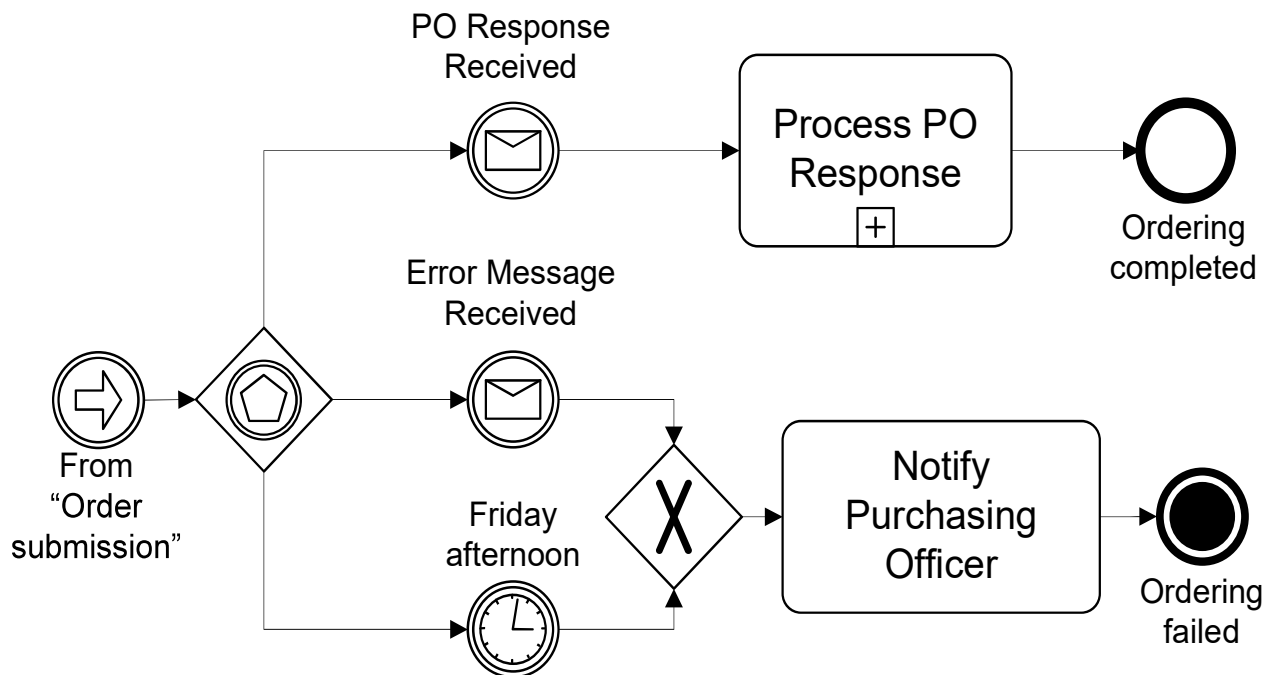
Eenvoudigste vorm van een exception: terminate event

- Exceptions zijn events die maken dat een process van zijn normale "koers" afwijkt
- Het *terminate end event* geeft enkel aan dat er een exception is.
- Het forceert dat het volledige process wordt afgebroken.



Voorbeeld: terminate event

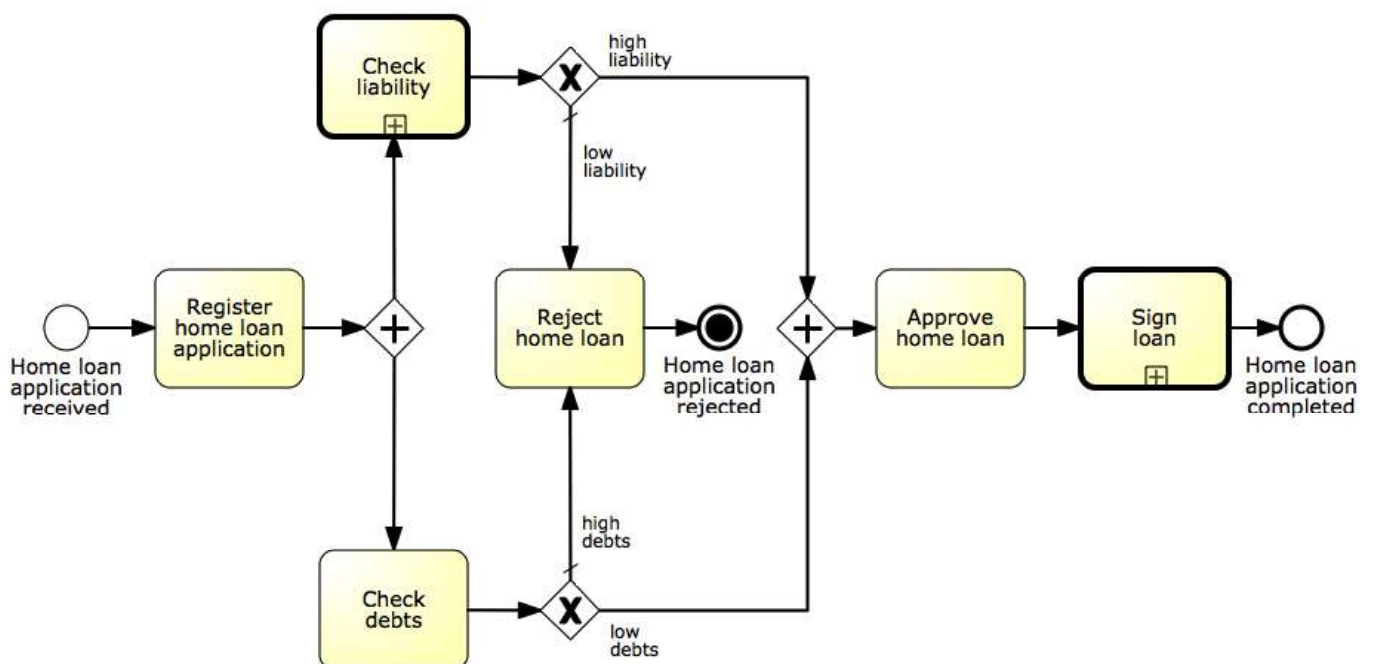
Signaleert de negatieve output.



Voorbeeld 2: terminate event

Nut van *terminate event*: het vermijdt deadlocks

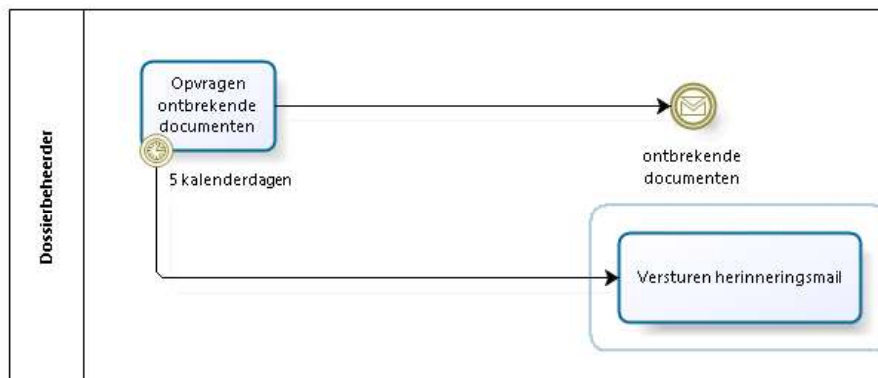
Want: de plus-gateway wacht tot alle taken "aangekomen" zijn.



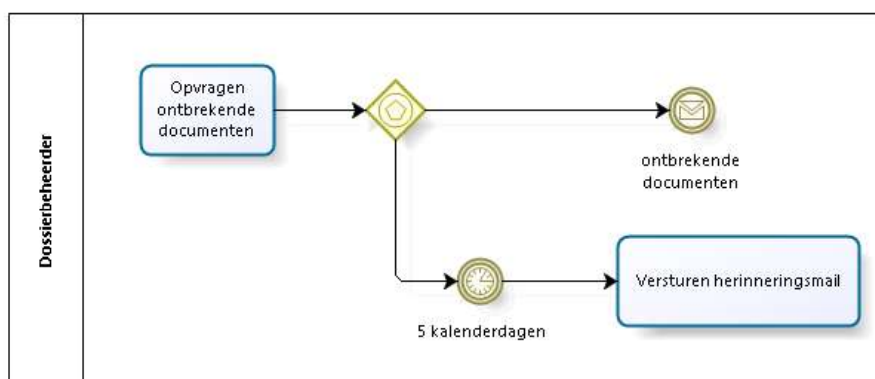
Uitzonderingsstromen

- Een **uitzonderingsevenement**
 - Gebeurtenis **tijdens de uitvoering** van het proces
 - De normale stroom van het proces wordt afgeleid buiten de normale stroom
 - Kunnen gegenereerd worden door een time-out, fout, boodschap,...
- Een **uitzonderingsstroom**
 - Sequentiële stroom die start met uitzonderings-evenement aan de grens van een activiteit
 - Enkel doorlopen wanneer uitzondering zich voordoet
 - Normale stroom wordt onderbroken

Uitzonderingsstroom



Is logisch equivalent met onderstaande, maar het bovenstaande beklemtoont de uitzonderingssituatie



Voorbeeld: ons orderverwerkingsproces

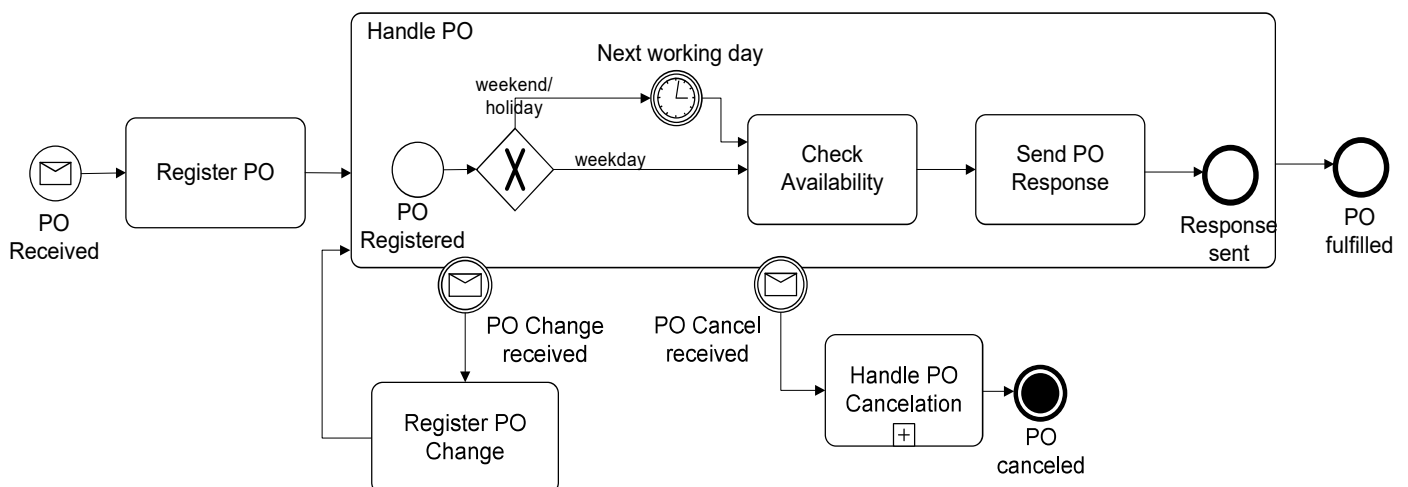
Het proces start als een PO (purchase order) binnenkomt. Eerst wordt het PO geregistreerd. Als de huidige dag geen werkdag is, wacht het proces tot de volgende werkdag om verder te gaan. Als het wel een werkdag is, wordt de beschikbaarheid gecontroleerd en er wordt een antwoord naar de klant gestuurd.

Op ieder moment ná registratie van het PO, kan de klant een "PO change request" sturen. Na ontvangst hiervan moet alle verwerking van het PO gestopt worden. De "PO change request" wordt dan eerst geregistreerd, waarna het process verder verloopt als bij een normale PO.

Als de klant een "PO cancellation request" stuurt na PO-registratie, moet de verwerking afgebroken worden.

41

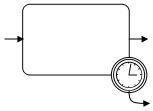
Oplossing



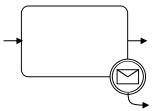
42

Exception handling

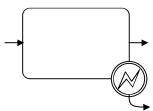
- Afhandelen van exceptions betekent meestal: taak/subproces stoppen en een speciale activiteit uitvoeren.
- Types van exceptions bij een activiteit (taak/subproces) in BPMN:



Timeout: de activiteit duurt te lang en moet onderbroken worden → afhandelen met een timer event.



Extern: er gaat iets mis buiten het process, en de uitvoering van de huidige activiteit moet onderbroken worden → afhandelen met een message event



Intern: er gaat iets mis in de activiteit zelf, waardoor de uitvoering wordt afgebroken → afhandelen met een error event

Al deze events onderbreken de omsluitende activiteit en starten de afhandeling

Interne exceptions: error event



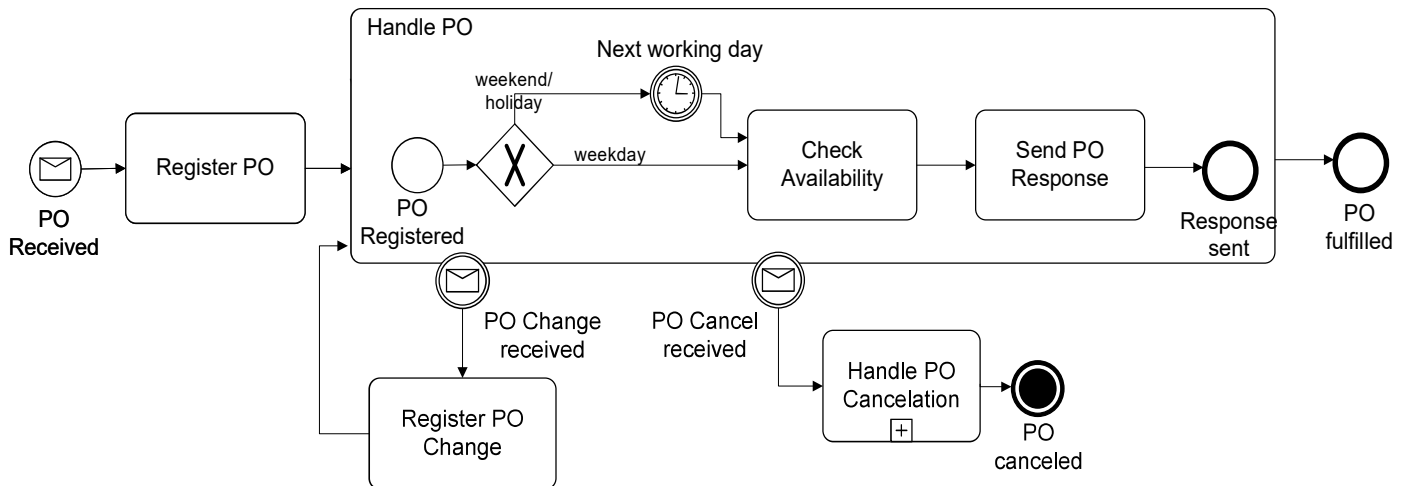
Error Event – geeft een error aan: de “eind”-versie (volle bliksem) genereert een error-event.

De “tussenliggende” versie wordt gehecht aan de grens van activiteit/subproces en verwerkt de uitzondering.

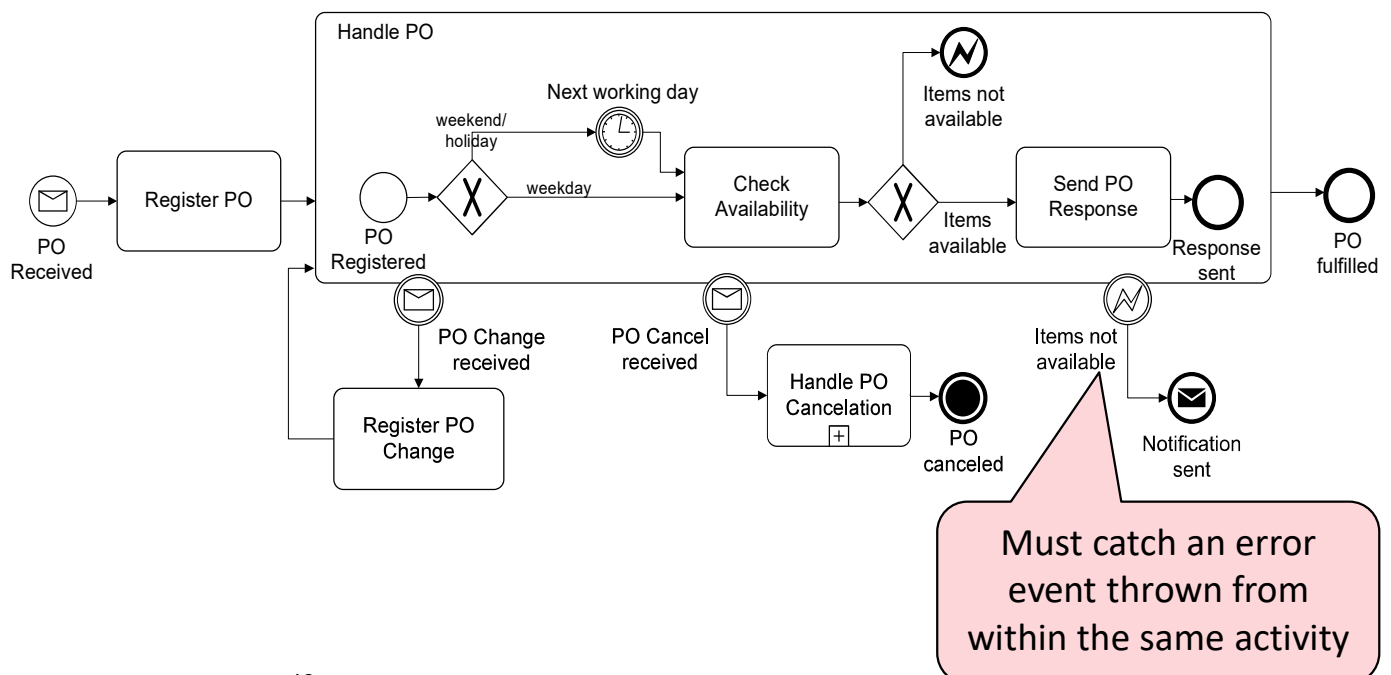


Voorbeeld: error event

Beschouw opnieuw het PO verwerkingsproces, met als uitbreiding: als een artikel niet beschikbaar is, dan wordt alle verwerking gestopt. De klant moet dan geïnformeerd worden dat het PO niet verder kan verwerkt worden.

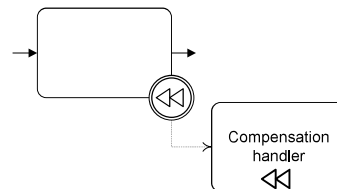


Oplossing



Compensation handling

- Kan gebruikt worden als onderdeel van een exception afhandelingsprocedure.
- = rollback van al afgeronde proces-activiteiten
bijv.: een boeking voor een reis is succesvol afgerond, maar dan blijkt dat de gegevens van de kredietkaart onvolledig waren. De boeking moet teruggedraaid worden.
- Getriggerd door het "gooien" van een compensatie event.
- Afgehandeld door *een compensation handler*



Compensate event



Compensate Event – Geeft aan dat het omvattende proces *gecompenseerd* moet worden: de “throwing intermediate” en de “eind”-versie genereren het compensate event, terwijl de “intermediate catching”-versie het “consumeert” als het gehecht is aan de grens van een activiteit of subproces.

Moet aan rand van activiteit hangen

Compensatie-event: geeft aan dat de omsluitende activiteit moet gecompenseerd worden.

3 versies:

- catching intermediate: start uitvoeren compensatie
- throwing intermediate: wordt opgevangen door catching intermediate
- throwing end

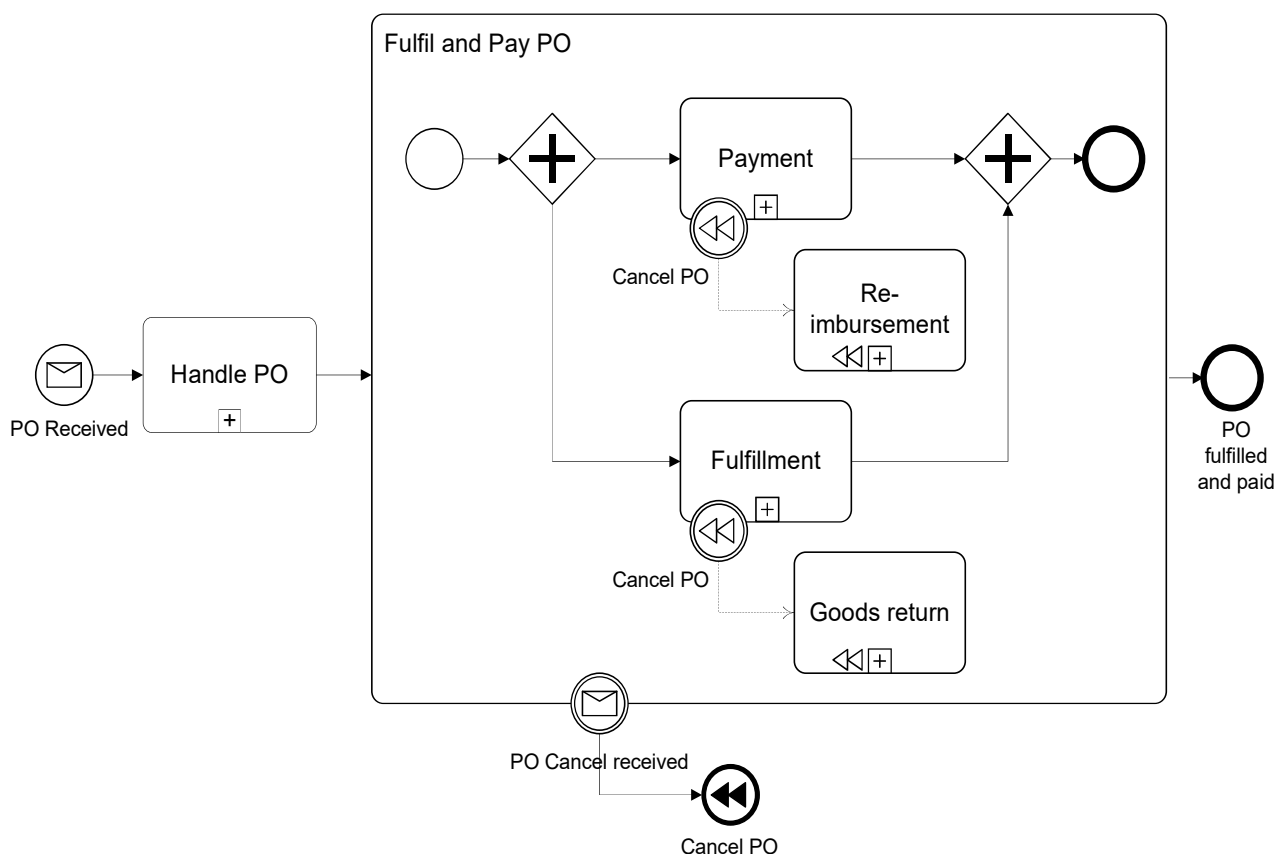
Voorbeeld: compensation event

Laten we opnieuw het PO-process uitbreiden.

Na registratie en check, worden een betalings- en een afhandelingssubproces in parallel gestart.

Tijdens deze 2 subprocessen kan een annulatie binnenkomen van de klant. In dat geval worden beide subprocessen gestopt, en een compensatie-routine wordt uitgevoerd (resp. terugbetaling en/of terugzending van goederen).

Mogelijke oplossing: compensation event



Herhaling: Terminate, Error and Compensate events

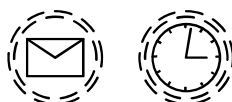
	Intermediate			End
	Catching	Boundary Interrupting	Throwing	
Terminate: Triggering the immediate termination of a process.				
Error: Catching or throwing named errors.				
Compensate: Handling or triggering compensation.				

Non-interrupting boundary events

Soms willen we een taak/subprocess triggeren in parallel met de normale flow ,dus zonder de normale flow te onderbreken.

Dit kun je modelleren met een *non-interrupting boundary event*.

- Rand in dubbele stippellijn
- Hangt aan de rand van een taak/sub-process



Voorbeeld: non-interrupting boundary events

Opnieuw het PO afhandelingsproces...

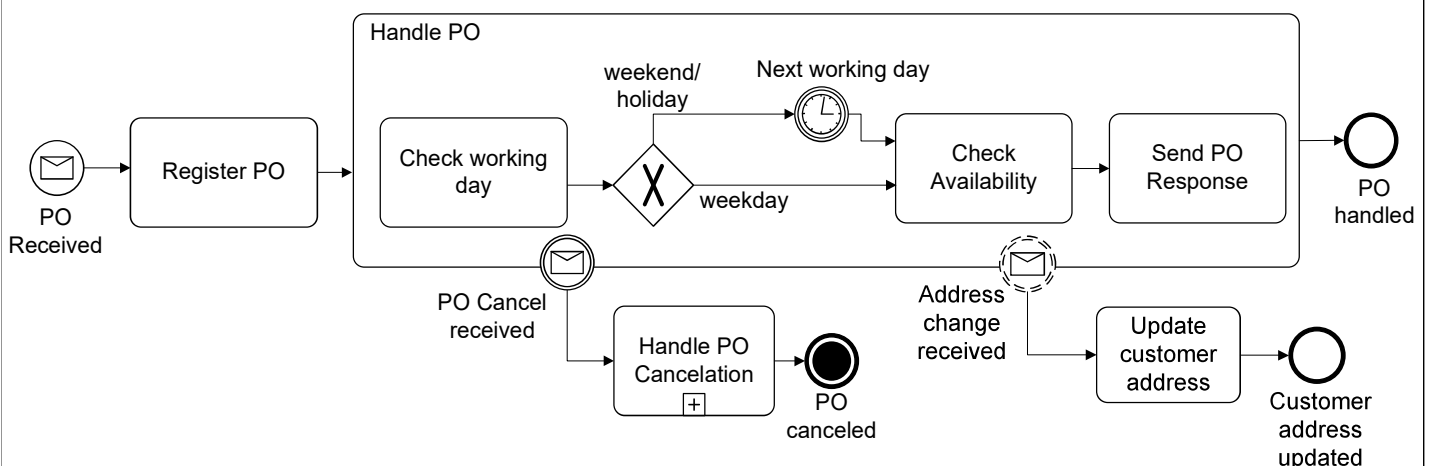
Het proces start als een PO (purchase order) binnenkomt. Eerst wordt het PO geregistreerd. Als de huidige dag geen werkdag is, wacht het proces tot de volgende werkdag om verder te gaan. Als het wel een werkdag is, wordt de beschikbaarheid gecontroleerd en er wordt een antwoord naar de klant gestuurd.

De klant kan een "Customer address change request" sturen na de PO registratie. Als zo'n aanvraag aankomt, wordt die geregistreerd zonder verdere actie.

Op ieder moment ná registratie van het PO, kan de klant een "PO change request" sturen. Na ontvangst hiervan moet alle verwerking van het PO gestopt worden. De "PO change request" wordt dan eerst geregistreerd, waarna het process verder verloopt als bij een normale PO.

Als de klant een "PO cancellation request" stuurt na PO-registratie, moet de verwerking afgebroken worden.

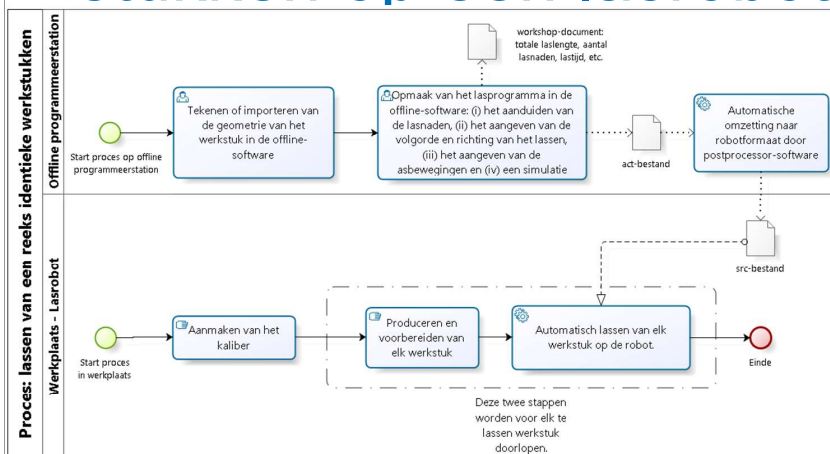
Oplossing: non-interrupting boundary events



Herhaling: Events

	Start		Intermediate		End	
	Top-Level	Event Sub-Process Interrupting	Catching	Boundary Interrupting	Throwing	
Untyped: indicate start point, state changes or final states.						
Message: Receiving and sending messages.						
Timer: Cyclic timer events, points in time, time spans or timeouts.						
Link: Off-page connectors. Two corresponding link events equal a sequence flow.						
Error: Catching or throwing named errors.						
Compensate: Handling or triggering compensation.						
Terminate: Triggering the immediate termination of a process.						

Oef. “lassen van een reeds identieke stukken op een lasrobot”



Terwijl het programma wordt opgesteld, worden in de werkplaats het kaliber (= een mal) gemaakt en worden de stukken voorbereid. Daarna kunnen de stukken gelast worden op de robot.

- Herbekijk de oplossing die eerder gegeven is en vorm om naar correcte BPMN met gebruik van alle mogelijkheden.

Oefening

Herwerk de pizza-oefening gebruik makend van exceptions. Kun je het nu nauwkeuriger modelleren?

Oefening

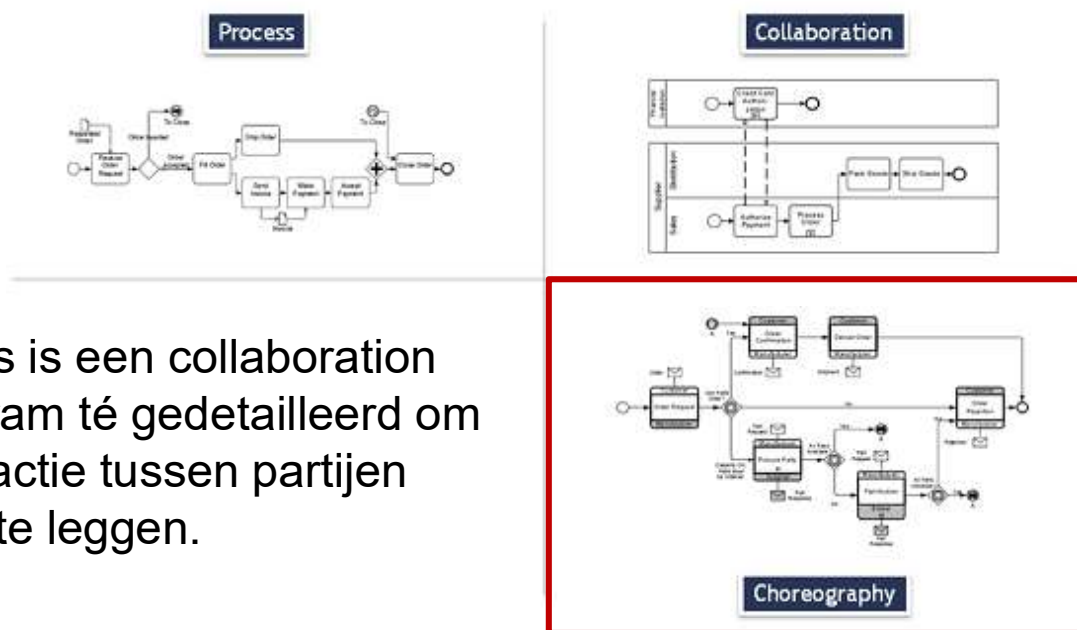
De procedure om in te loggen bij een Internet bankrekening start als de user zijn login-gegevens heeft ingegeven. Eerst wordt de usernaam gevalideerd. Als die geldig is, wordt het aantal paswoord-pogingen op 0 gezet en wordt het paswoord gevalideerd. Als dit ongeldig is, wordt de teller van het aantal pogingen verhoogd en als dat < 3 wordt de user gevraagd nogmaals zijn paswoord in te geven, ditmaal samen met een CAPTCHA-test. Bij 3 mislukte pogingen, wordt de procedure afgebroken en de rekening geblokkeerd. De validatie-procedure wordt afgebroken als de validatie-server niet beschikbaar is tijdens validatie. De captcha-test wordt afgebroken als de captcha-server niet beschikbaar is op dat moment. In beide gevallen wordt de user geïnformeerd. Op elk moment tijdens de procedure kan de user de webpagina sluiten, wat ook in het afbreken van de procedure resulteert.

Oef. Groothandelaar en transporteurs

Elke week, tegen vrijdagavond, moet groothandelaar Peersman producten leveren bij tientallen verschillende klanten. Hij stuurt hiervoor telkens op maandagmorgen een offerte-aanvraag naar 5 transporteurs, die met een offerte moeten antwoorden tegen woensdagmiddag. Doen deze dat niet, dan stuurt de groothandelaar een herinnering. Om een offerte te kunnen opstellen moet elke transporteur de ideale route berekenen en het gebruik van de vrachtwagen inschatten (vol, halfvol, quasi leeg, ...). Op donderdagmiddag beslist de groothandelaar met wie hij in zee gaat en plaatst de bestelling. De overigen krijgen een bericht van afwijzing.

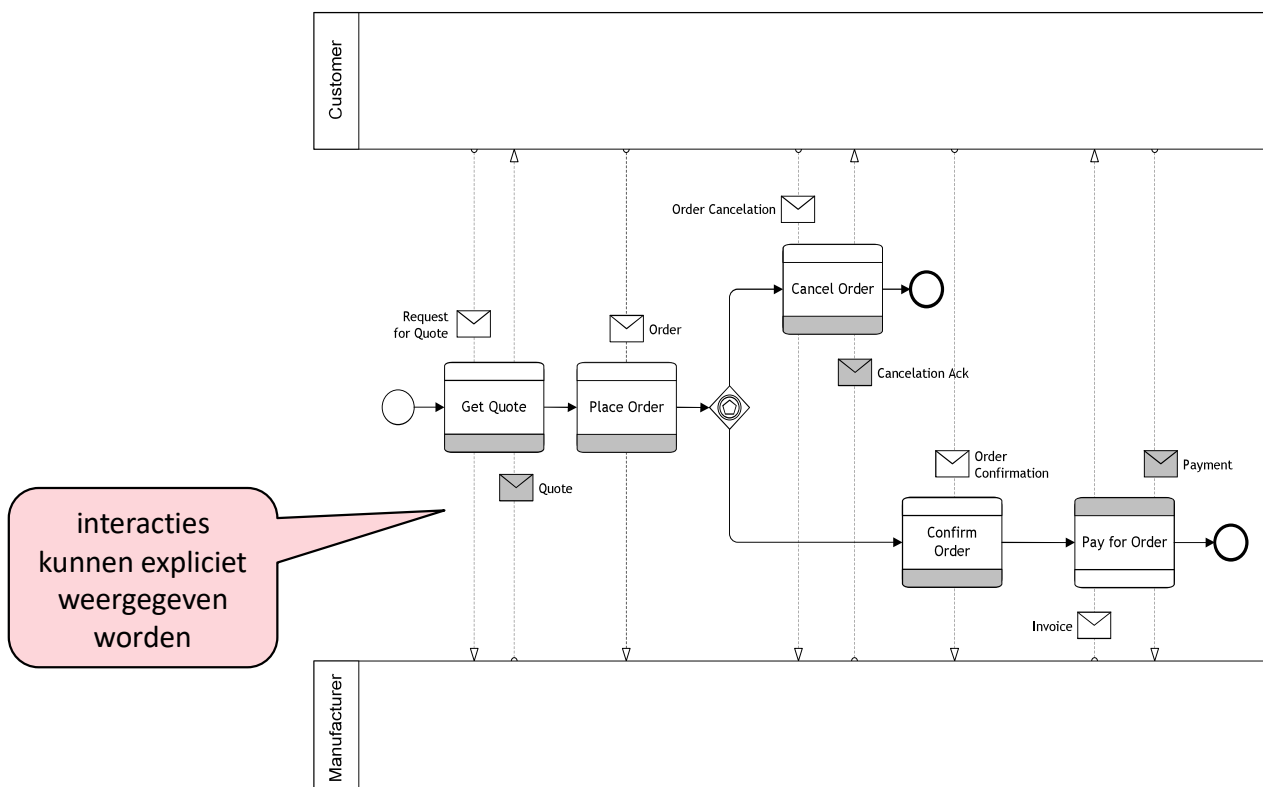
Choreografieën

BPMN 2.0 model types

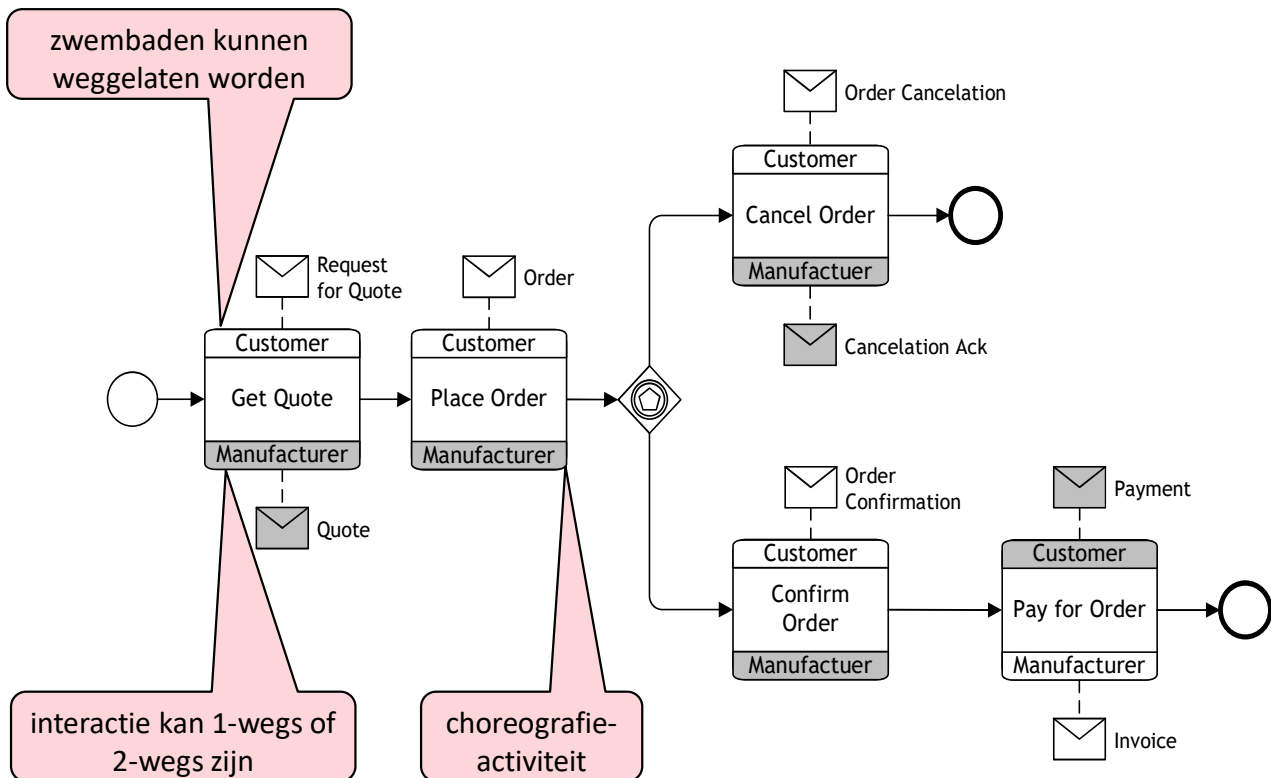


Soms is een collaboration diagram té gedetailleerd om interactie tussen partijen vast te leggen.

Choreografie-diagram



Choreografie-diagram (vervolg)



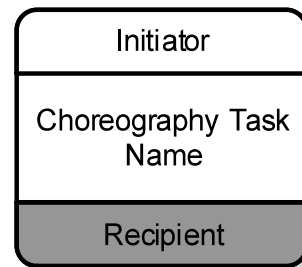
Choreografie

Procesmodel van de interacties die plaatsvinden tussen 2 of meer business-partijen

- Focust op uitwisseling van boodschappen tussen partijen
- Geldt als contract tussen partijen
- Kan verfijnd worden in privé-processen of in een collaboration diagram.

Choreografie-taak

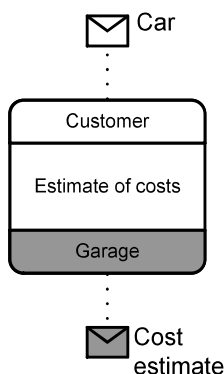
- Atomische activiteit
- Stelt een interactie voor tussen 2 business partijen
- Ofwel één-wegs (asynchroon), ofwel twee-wegs (synchroon)
- Onderscheid tussen sturende en ontvangende partij



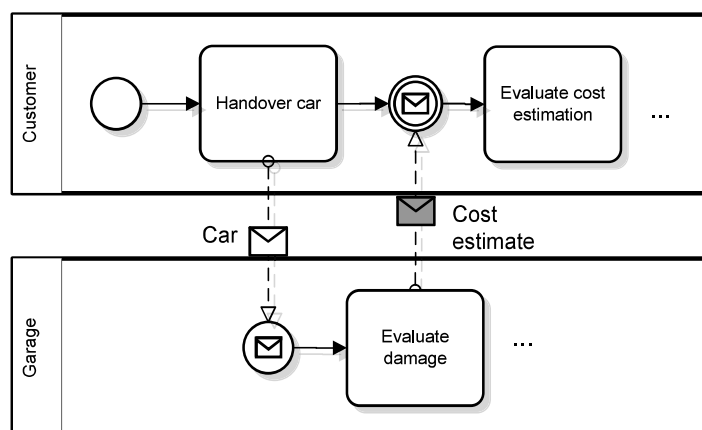
Choreografie-taak

- Strook van initiërende partij niet opgevuld
- Boodschap-icons zijn optioneel

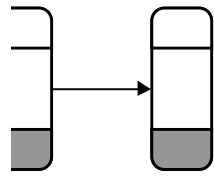
Choreografie-taak



Equivalent collaboration-diagram



Basic Choreografie-elementen



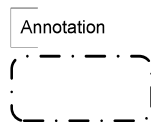
Sequence flow – Connects and orders choreography tasks, events and gateways



Events – Most process events allowed. No non-interrupting events



Gateways – All process gateways allowed: Exclusive, Inclusive, Parallel and Event-based gateways

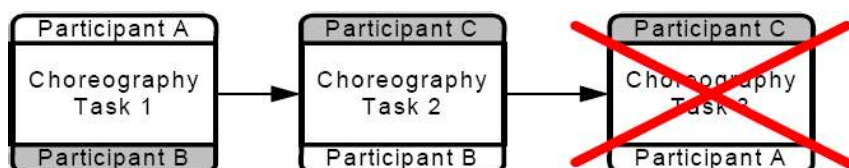


Text annotations, groups – No restrictions on their use

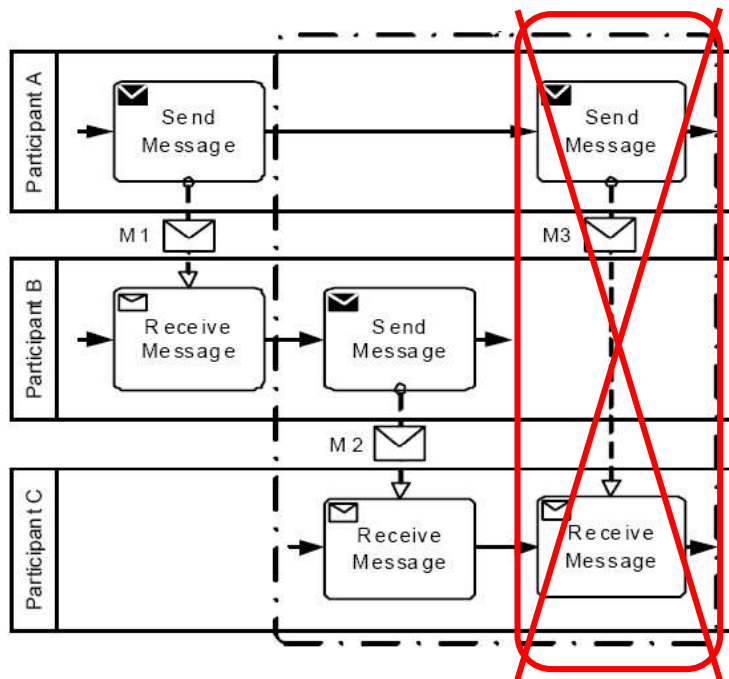
Choreografie-volgordebependingen

De initiator van een choreografie-activiteit moet betrokken zijn in de vorige activiteit (met uitzondering van de 1ste activiteit)

Waarom ?!



Choreografie-volgordebepeningen

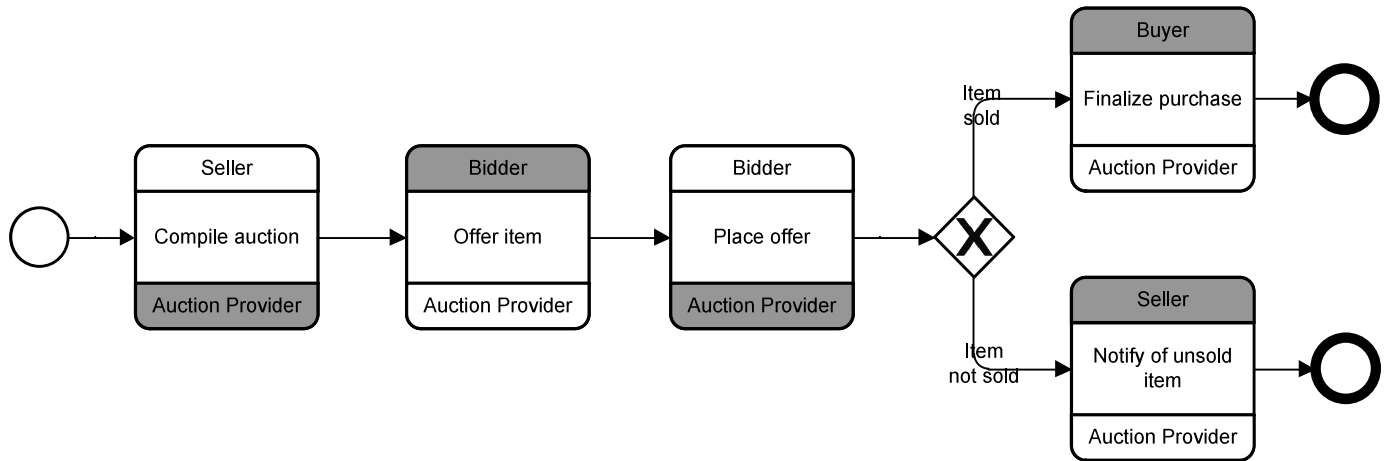


Business Process Model and
Notation, Page 339
OMG Document Number:
formal/2011-01-03
January 2011

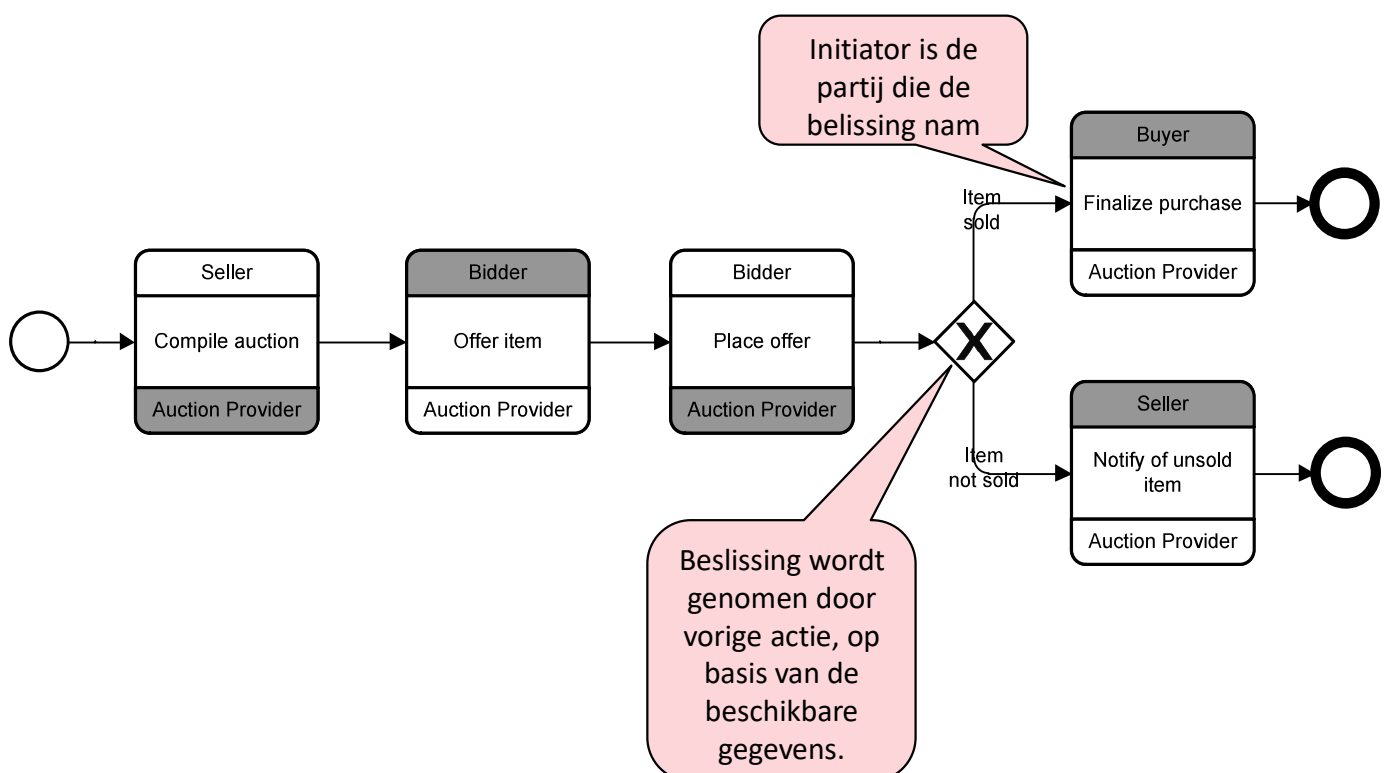
Voorbeeld: Choreografie van een veiling

Een verkoper stuurt informatie over een artikel dat hij wil verkopen naar een veilingorganisator. De organisator publiceert de veiling door het artikel aan te bieden aan een bidder. Als de veiling gestart is, kan de bidder een bod plaatsen. Als het artikel verkocht is, dan zal de organisator de verkoop afronden met de koper, zoniet informeert hij de verkoper dat het artikel niet verkocht werd.

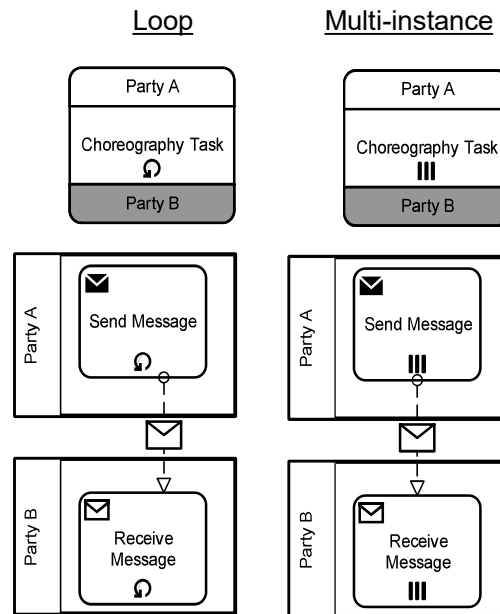
Oplossing: Choreografie van een veiling



Oplossing: Choreografie van een veiling

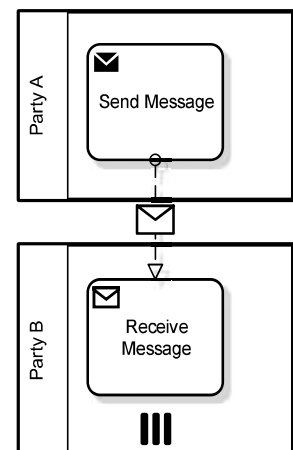
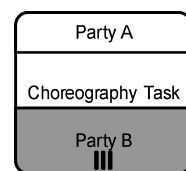


Choreografie-taak – Interne *markers*

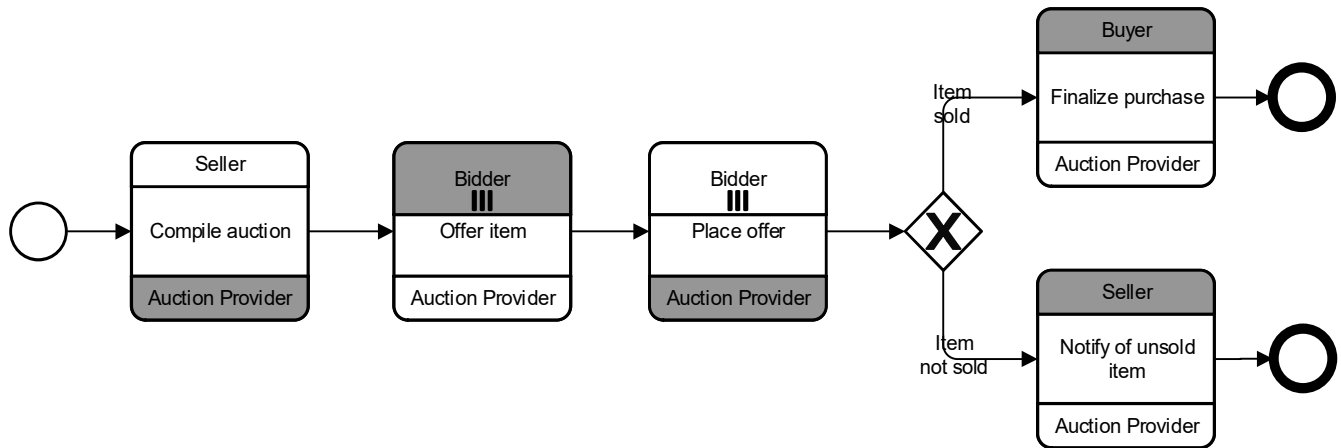


Choreografie-taak Multi-instance Party marker

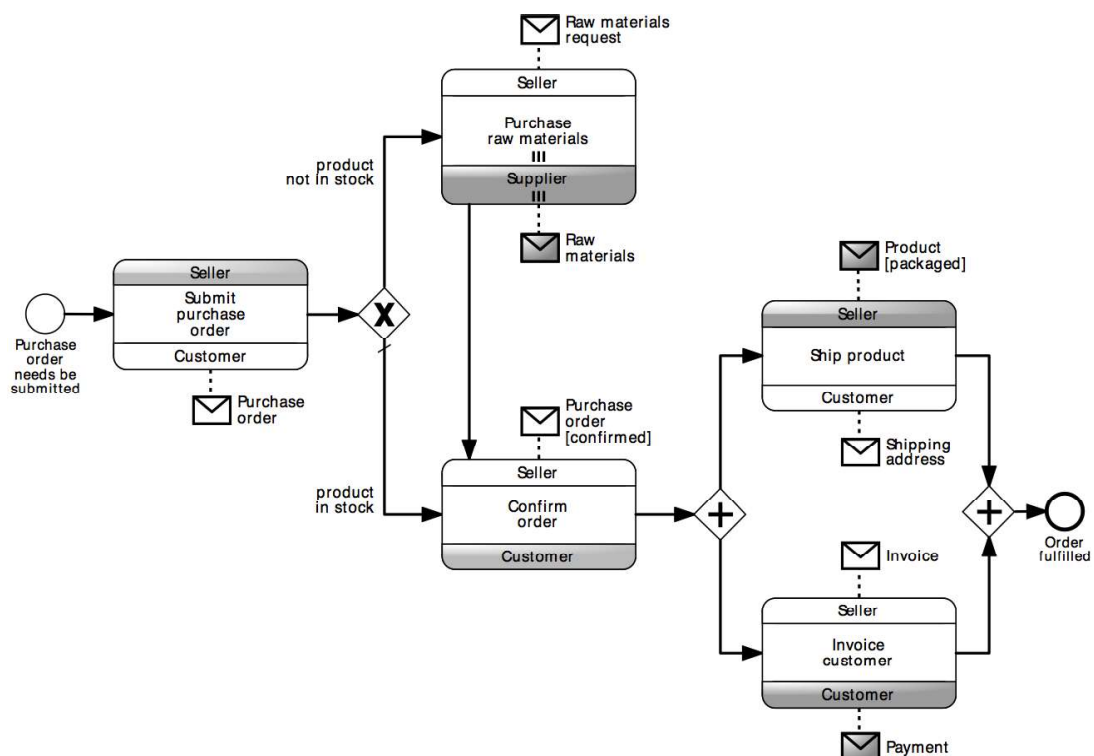
- Partijen kunnen multi-instance zijn
- bijv. klanten of leveranciers



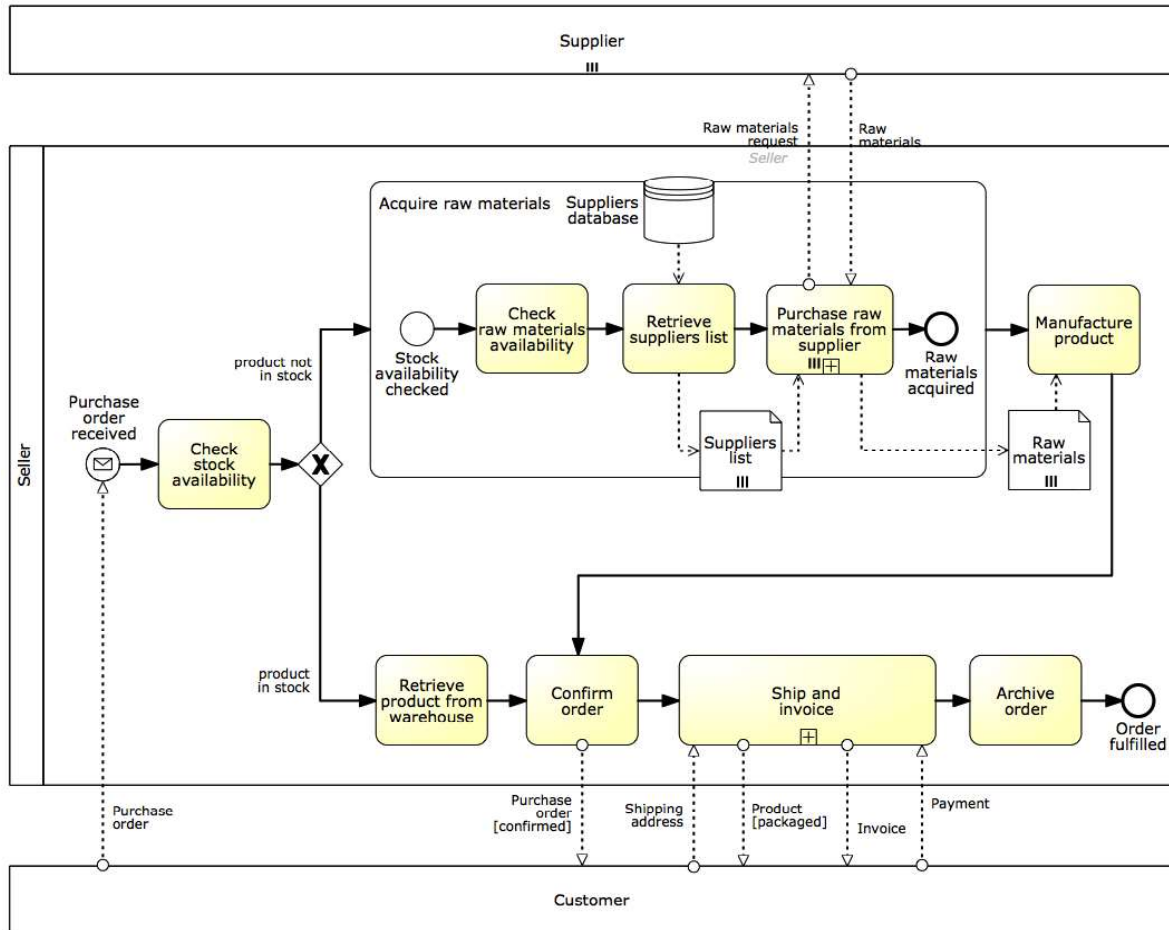
Voorbeeld: Choreografie van een veiling met multi-instance



Voorbeeld: Choreografie



Zelfde proces als collaboration-diagram



Dus wat is het verschil tussen een choreografie-diagram en een collaboration-diagram?

Oefening

- Zet de choreografie van een veiling (met en zonder multi-instance) om naar een collaboration-diagram