

Upravljanje poslovnim procesima (UPP)

Nositelj: izv. prof. dr. sc. Darko Etinger

Asistent: Luka Blašković, mag. inf.

Ustanova: Sveučilište Jurja Dobrile u Puli, Fakultet informatike u Puli



Fakultet informatike u Puli

(3) Složena grananja



Skretnice određuju redoslijed izvođenja aktivnosti u procesu. Do sada smo vidjeli kako jednostavne ekskluzivne skretnice omogućuju odabir između dvije ili više opcija tijekom izvođenja poslovnog procesa. Važno je naglasiti da se pritom uvijek odabire samo jedna opcija — ona koja ispunjava uvjet. U ovom ćemo poglavlju upoznati i druge vrste skretnica, uključujući paralelnu i inkluzivnu. Također, upoznat ćemo se s načinima definiranja „čekanja“ na rezultate aktivnosti koje ovise o više uvjeta, koristeći različite oblike spajanja i grananja sljedova aktivnosti.

 Posljednje ažurirano: 9.11.2025.

Sadržaj

- [Upravljanje poslovnim procesima \(UPP\)](#)
- [\(3\) Složena grananja](#)
 - [Sadržaj](#)
- [1. Ekskluzivna \(eng. Exclusive\) skretnica](#)
 - [1.1. **XOR** skretnica spajanja \(eng. XOR merge/join gateway\)](#)
- [2. Paralelna \(eng. Parallel\) skretnica](#)
 - [2.1. **AND** skretnica spajanja \(eng. AND merge/join\)](#)
- [3. Inkluzivna \(eng. Inclusive\) skretnica](#)
- [4. Ukratko, kada koristiti koju skretnicu?](#)
- [Zadaci za Vježbu 3](#)
 - [1. Wolt - dostava hrane](#)
 - [2. Liječenje pacijenta koji dolazi na hitnu pomoć](#)
 - [3. Organizacija koncerta u Pulskoj Areni](#)

1. Ekskluzivna (eng. Exclusive) skretnica

Ekskluzivnu (XOR) skretnicu (eng. *Exclusive gateway*) već ste upoznali kroz prethodne primjere. Ona se koristi za odabir jedne između više opcija, gdje se uvijek odabire samo maksimalno jedna opcija. Ukoliko je zadovoljen predikat (rezultat poslovne aktivnosti i/ili poslovna odluka) definiran na skretnici, izvršava se **samo jedan slijedni tok** dok se ostali tokovi zanemaruju.



Ekskluzivna skretnica se definira romбом, s oznakom **x**

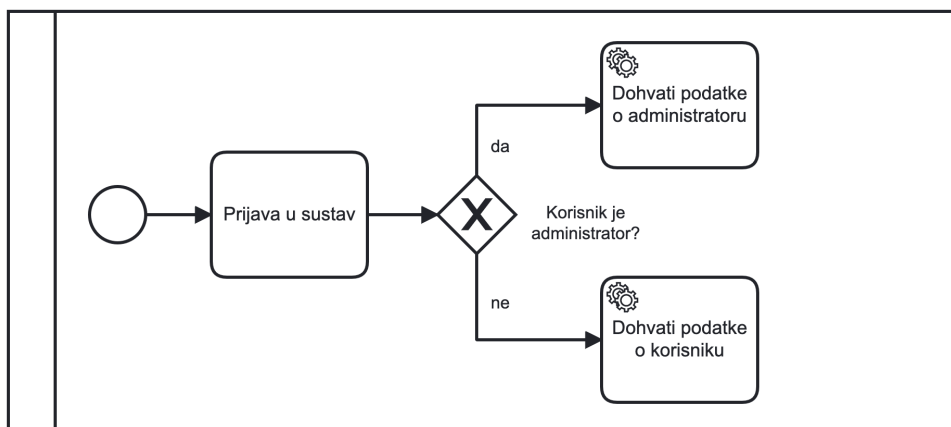
Primjer ekskluzivne skretnice: Nakon što se korisnik prijavi na sustav, želimo provjeriti je li korisnik administrator ili običan korisnik. Ukoliko je korisnik administrator, želimo mu prikazati dodatne opcije koje običan korisnik nema. Ukoliko je običan korisnik, želimo mu prikazati standardne opcije, poput pregleda profila, postavki i sl.

Samim time, na XOR skretnicu ćemo upisati uvjet `je administrator?` ili `korisnik je administrator?`. Iz skretnice definiramo **dva ili više toka** koristeći sekvencijalne tokove (obične strelice) prema aktivnostima koje želimo izvršiti. **XOR skretnica je u pravilu skretnica uvjetovana podacima koje sa sobom donosi instanca procesa** (npr. korisnička uloga, tip korisnika, itd.). Ali, može biti i složena odluka bazirana na više uvjeta (ovo ćemo modelirati na budućim vježbama kroz DMN notaciju).

Opisani način korištenja skretnice zovemo **grananje** (eng. *splitting/branching*).

Skretnica grananja (ili samo grananje) ima jedan ulazni slijedni tok ili ulazni put (grafički uobičajeno usmjeren u lijevi ili gornji kut romba), a više izlaznih sljedova (iz desnog ili donjeg kuta romba).

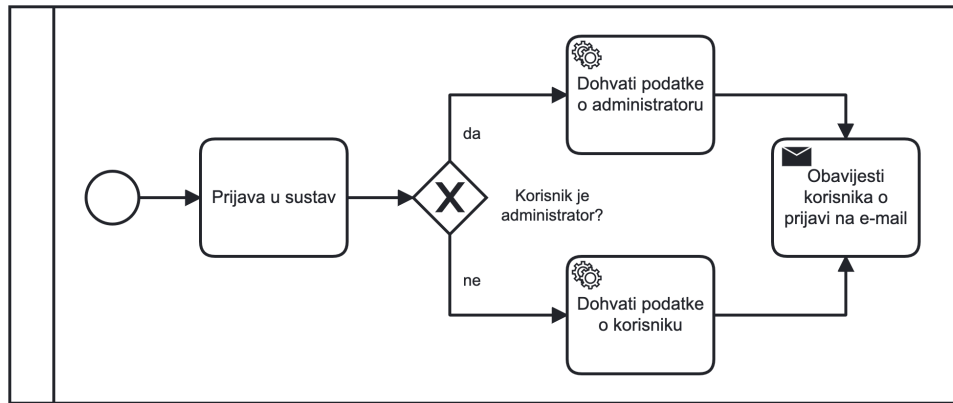
Grananjem definiramo **više izlaznih tokova**, ali se **izvršava samo jedan tok** - onaj za koji je uvjet zadovoljen.



Slika 1: Primjer korištenja XOR skretnice za odabir između vrste korisnika

Međutim, što ako je sljedeća aktivnost koja slijedi nakon odabira jednaka za oba korisnika? U tom slučaju, ne želimo ponavljati istu aktivnost za svaku opciju, već želimo samo strelice toka preusmjeriti u ponavljajuću aktivnost.

Primjer: Nakon prijave i dohvata podataka o korisniku, želimo poslati korisniku obavijest o uspješnoj prijavi na e-mail.



Slika 2: Nakon odabrane opcije, jednostavno prosljeđujemo tok prema sljedećoj aktivnosti

Iako ovakva sintaksa nije pogrešna, prema BPMN standardu preporuka je koristiti ekvivalentnu XOR skretnicu spajanja (*eng. XOR merge gateway*) kako bismo naglasili da se svi tokovi ponovno spajaju u jedan tok. Ovo je posebno važno kod složenijih modela procesa gdje se spajaju dolazni tokovi iz više različitih grananja.

Ipak, u jednostavnim slučajevima poput ovog gdje se samo 2 toka "spajaju" u jednu aktivnost, možemo definirati i bez skretnice spajanja.

1.1. XOR skretnica spajanja (eng. XOR merge/join gateway)

Osim grananja, koje smo do sad koristili, skretnice je moguće koristiti i za **spajanje** (eng. *merge/join*). Spajanje se koristi kada se više tokova treba spojiti u jedan, tj. kada se više tokova vraća u jedan tok. Drugim riječima, **skretnica spajanja ima više ulaznih slijednih tokova** (npr. *lijevo, gore, dolje*) i **jedan izlazni slijedni tok** ili izlazni put (*desno*).

Dobra je praksa koristiti XOR skretnicu spajanja **kako bi naglasili da se izlazni tokovi spajaju u jedan tok**, koji se nastavlja čim je zadovoljen uvjet.

Kod XOR skretnice, **čak i kad je moguće da je više uvjeta istinito, ona aktivnost koja prva završi će nastaviti tok** (budući da je XOR skretnica ekskluzivna - uvjeti se isključuju). Dakle, iako je moguće definirati inkluzivne aktivnosti pred ovu skretnicu, to nije poželjno.

Ako se vratimo na primjer iznad, XOR merge skretnicu jednostavno postavljamo prije same aktivnosti i **ne dodajemo joj naziv!**

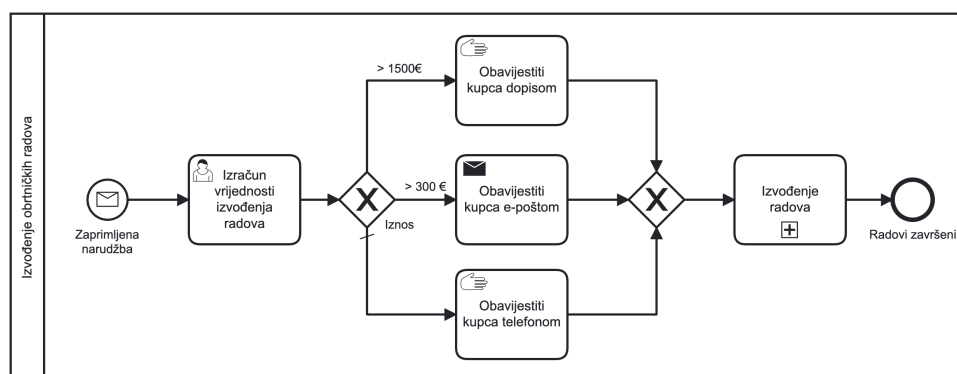
Slika 3: Dodajemo XOR merge skretnicu u koju se "spajaju" svi izlazni tokovi XOR split skretnice

XOR skretnicu spajanja interpretirajte kao: "pričekaj ulaz barem jednog toka, a zatim nastavi dalje". U gornjem slučaju: "pričekaj dohvat podataka o korisniku (bilo da se radi o administratoru ili korisniku), a zatim obavijesti korisnika o uspješnoj prijavi".

Rekli smo da se **XOR skretnicom može definirati i više od dva izlazna toka** (recimo kad predikat nije Boolean oblika, ili aktivnost ima više od 2 moguća ishoda). Takva sintaksa je dozvoljena i onda je **vrlo poželjno koristi ekvivalentnu XOR skretnicu spajanja**.

Primjer: Vlasnik tvrtke za izvođenje obrtničkih radova dobiva narudžbu. Obrtnik će izračunati vrijednost radova i obavijestiti naručitelja **telefonom, mailom** ili **dopisom**, ovisno o vrijednosti, a tek onda izvesti naručeno.

- **XOR skretnica grananja** za odabir načina obavješćavanja (**telefon, mail, dopis**) ovisno o podacima (**vrijednost izvođenja radova**)
- **XOR skretnica spajanja** za spajanje toka nakon obavješćavanja naručitelja i nastavak prema potprocesu izvođenja radova



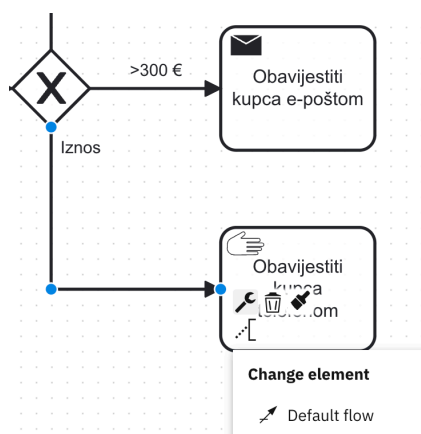
Slika 4: XOR skretnica grananja za odabir načina obavješćavanja ovisno o vrijednosti radova. Odabrani tok zatim izvršava pojedinu aktivnost, a rezultat se spaja kroz XOR skretnicu spajanja prije potprocesa "Izvođenje radova".

Definirali smo 3 moguća toka ovisno o iznosu:

- **Slanje dopisa:** za iznose veće od 1500 eura
- **Slanje e-maila:** za iznose veće od 300 eura
- **Obavješćavanje telefonom:** za iznose manje od 300 eura

Logičke izraze (`> 1500 eur` i `> 300 eur`) smo zapisali na strelicama, a oznakom (`/`) smo definirali zadani (*defaultni*) tok (za iznose manje od 300 eura).

Default flow predstavlja zadani sljed nakon skretnice koji će se izvršiti ukoliko nijedan od definiranih uvjeta nije zadovoljen (niti jedan drugi uvjet se ne evaluiira u istinitu vrijednost). Moguće ga je definirati za OR i XOR skretnice.



Slika 5: Odabir zadanog toka (*eng. Default flow*) iz XOR skretnice grananja. Iako je moguće definirati uvjet na zadanom toku (`< 300 €`), preporuka je zadani tok (ako postoji) označiti bez uvjeta radi brže čitljivosti modela procesa.

Nakon obavješćavanja, ovisno o uvjetu odabrat će se samo jedna aktivnost, a skretnica spajanja će **pričekati na ulazni tok jedne od aktivnosti i nastaviti dalje**.

2. Paralelna (eng. Parallel) skretnica

Paralelna (AND) skretnica (eng. *Parallel gateway*) koristi se za modeliranje situacija u kojoj se **tok procesa grananja dešava paralelno**, odnosno kada želimo definirati više aktivnosti ili sekvencijalnih tokova koji se izvršavaju "istovremeno" (paralelno).

Pri grananju (*splitting/branching*) AND skretnica aktivira sve izlazne sljedove "istovremeno".

Pri spajanju (*merging/joining*) pokreće izlazni sljed tek kada su svi ulazni sljedovi završeni. (**VAŽNO!**)



Paralelna skretnica se definira rombom, ali s oznakom +

Napomena: Za razliku od XOR skretnice, kod **paralelne skretnice** sve aktivnosti koje slijede nakon nje započinju se izvršavati **paralelno** - ipak, to ne znači da će sve aktivnosti završiti u isto vrijeme, niti će trajati isto dugo, već će se započeti izvršavati u isto vrijeme.

Paralelna skretnica spajanja povezuje se s logičkom konjunkcijom (operator **AND**) na sljedeći način:

- Kod grananja, ponaša se kao *broadcasting* - svi izlazni tokovi se aktiviraju istovremeno.
- Kod spajanja, ponaša se kao *synchronization* - izlazni tok paralelne skretnice spajanja se aktivira tek kada su svi ulazni sljedovi završeni.

Međutim, u stvarnim scenarijima, aktivnosti koje slijede nakon paralelne skretnice gotovo nikad nisu istog trajanja. Neke mogu trajati nekoliko minuta, više sati pa i nekoliko dana.

Ono što je ključno, jest da se sve aktivnosti **započinju izvršavati istovremeno**, ali sekvencijalni tok koji nastavlja izvršavanje procesa nastavlja **tek onda kada su sve aktivnosti završene**. Drugim riječima, moramo prikazati **čekanje na izvršavanje svih aktivnosti** koje slijede nakon paralelne (AND) skretnice.

Slika 6: Paralelna (AND) skretnica za definiranje paralelnog grananja u 2 slijeda aktivnosti

AND skretnice u pravilu **ne želimo imenovati** budući da se sve aktivnosti koje slijede započinju bezuvjetno izvršavati, odnosno nema predikata kojim se uvjetuje izbor između aktivnosti.

Ipak, moguće je dodati naziv ako želimo dodatno pojasniti što se događa (npr. "Pokretanje svih internih procesa", "Paralelna obrada podataka", "Početak sveobuhvatne provjere kvalitete robe",

Napomena: U definiciji AND skretnice navedeno je da se aktivnosti izvršavaju paralelno, odnosno "istovremeno". Namjerno je napisano pod navodnim znakovima budući da se aktivnosti u stvarnosti ne izvršavaju sinkronizirano istovremeno, već pseudoparalelno, odnosno **konkurentno**. Što to znači? Više zadataka je pokrenuto istovremeno i u tijeku su njihova izvršavanja, ali se izvršavanje svakog zadatka odvija u vlastitom vremenskom okviru i **ne ovisi o izvršavanju drugih zadataka** (nije istovremeno s ostalima).

U nastavku ćemo vidjeti kako definirati čekanje na izvršavanje svih aktivnosti kroz paralelnu skretnicu spajanja.

2.1 AND skretnica spajanja (eng. AND merge/join)

Kako ćemo definirati čekanje na izvršavanje svih aktivnosti?

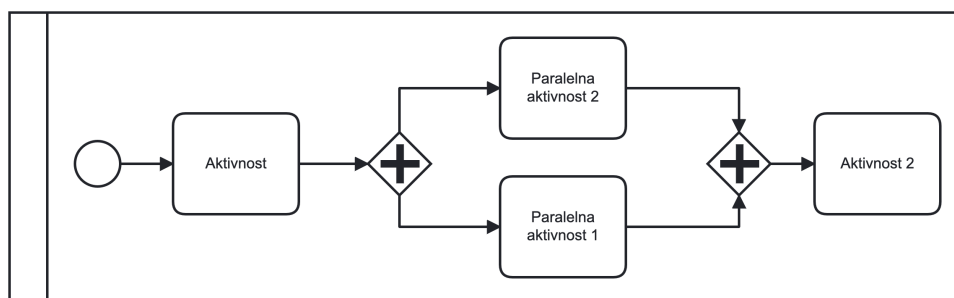
Česta pogreška bila bi jednostavno povezivanje sekvencijalnim tokom sve aktivnosti koje slijede nakon paralelne skretnice. To nije ispravno jer bi na taj način definirali da se aktivnosti izvršavaju sekvencijalno, bez čekanja na izvršavanje svih.

Slika 6: **Pogrešan način** povezivanja sljedova aktivnosti nakon paralelne skretnice grananja

Ono što ustvari moramo je definirati spajanje svih tokova kroz **paralelnu skretnicu spajanja** (eng. *parallel merge gateway*). Preciznije, **želimo prikazati čekanje na izvršavanje svih aktivnosti** kroz ekvivalentnu skretnicu spajanja.

Samim time, kod korištenja paralelnih skretnica za grupiranje dolaznih sljedova u jedan, prema BPMN standardu, **obavezno je definirati ekvivalentnu paralelnu skretnicu spajanja**.

Ispravno je sljedeće:



Slika 7: **Ispravan način** povezivanja toka nakon paralelne skretnice koristeći **AND skretnicu spajanja**

Primjer korištenja paralelne skretnice kod procesa obrade narudžbe:

Imamo web shop i želimo definirati slijed aktivnosti nakon što korisnik napravi narudžbu. Primjerice, jednom kad zaprimimo email s novom narudžbom, želimo poslati korisniku automatsku obavijest o zaprimljenoj narudžbi i paralelno provjeriti uplatu. Odnosno, potvrda se šalje automatski kroz neki servis koji smo integrirali, dok se provjera uplate obavlja kroz IS.

Iskoristit ćemo AND skretnicu kako bi prikazali paralelno slanje obavijesti (potvrde) i provjeru uplate u IT sustavu.

Slika 8: Primjer korištenja AND skretnice za paralelno slanje obavijesti i provjeru uplate

Ovdje paralelno izvršavamo 2 zadatka (premda ih može biti proizvoljan broj, nije loše ograničiti na najviše 3-4 radi preglednosti modela):

1. **Slanje potvrde o narudžbi korisniku** (*Send Task*)
2. **Provjera uplate** (*User Task*)

Koliko će se izvršavati svaka aktivnost?

- **Slanje potvrde o narudžbi korisniku** - automatski, bez čekanja ako je sustav ispravan, cca. 4-5 sekundi
- **Provjera uplate** - ovisno o načinu plaćanja, može trajati odmah (ako je plaćeno karticom) do nekoliko dana (npr. ako je uplaćeno putem inozemne bankovne transakcije)

Potrebno je **pričekati na izvršavanje svih aktivnosti** prije nastavka potprocesa "Otprema proizvoda" - upravo to prikazujemo AND skretnicom spajanja.

Međutim, što ako uplata nije uspješna? Nema problema, **možemo kombinirati AND skretnicu s XOR skretnicom** kako bismo definirali alternativni tijek aktivnosti.

Slika 9: Primjer kombiniranja AND i XOR skretnice za definiranje alternativnog slijeda aktivnosti

*Primjer korištenja paralelnih skretnica spajanja i grananja u procesu **produljenja registracije motornog vozila**. Kako je proces dovoljno poznat i ne treba ga posebno objašnjavati, prikazat ćemo samo dijagram gdje je istaknuto da se **dokumenti za registraciju mogu predati ako je prije toga: (1) uspješno obavljen tehnički pregled i (2) uplaćeno osiguranje vozila.***

Slika 10: Proces produljenja registracije motornog vozila. Entiteti na sekvencijalnim vezama nisu obavezni, ali pomažu u razumijevanju procesa.

Iako je moguće aktivnosti "Uplatiti osiguranje za vozilo" i "Obaviti tehnički pregled" prikazati kao sekvencijalne, **AND skretnicama želimo naglasiti dvije stvari:**

1. **da se sljedovi aktivnosti izvode paralelno** (bez sekvencijalne ovisnosti između njih), i
2. **da je potrebno izvršiti oba slijeda aktivnosti** prije nego je motorno vozilo moguće registrirati.

Možete još uočiti različite entitete: "Prometna dozvola", "Stara polica osiguranja", "Nova polica osiguranja" i "Potvrda o ispravnosti vozila" koje smo definirali na sekvencijalnim sljedovima.

Entitete (*eng. Entities*) je moguće definirati na informacijskim tokovima (*eng. message flow*) kako bi dodatno pojasnili koje se informacije razmjenjuju između sudionika u procesu, ali i na sekvencijalnim tokovima primjerice kako bi prikazali koje se informacije ili resursi koriste (npr. dokumenti, podaci, materijali i sl.) tijekom izvođenja aktivnosti.

Entiteti nisu obavezni, ali mogu pomoći u boljem razumijevanju procesa.

Podsjetnik: Sekvencijalni tokovi (*eng. Sequence flow*) se koriste za povezivanje elemenata unutar jednog procesa i označavaju se punim strelicama, dok se informacijski tokovi (*eng. message flow*) koriste za povezivanje elemenata između različitih sudionika (polja) i označavaju se isprekidanim strelicama.

Ima li smisla koristiti AND skretnicu kada nam aktivnosti imaju različita trajanja? Ukratko - DA, ima smisla. Rekli smo da je razlog taj što želimo prikazati da aktivnosti **započinju paralelno - split** (nema očite međuovisnosti između njih) i da je potrebno **pričekati na izvršavanje svih aktivnosti - merge**, prije nastavka procesa.

Primjer: U procesu prijave preferencija na studentsku praksu, traži se odabir najmanje 3 ponuđene opcije (zadataka). Nakon odabira, započinje se paralelno izvođenje tri različita slijeda aktivnosti: **1. Obavješćavanje poslodavca i evaluacija kandidata kod poslodavca, 2. Studentska priprema za evaluaciju, i 3. Pohrana odabira preferencija u bazu podataka i obavješćavanje studenta o odabranom zadatku.**

Slika 11: Proces prijave preferencija na studentsku praksu s paralelnom AND skretnicom grananja i spajanja

U ovom konkretnom primjeru, **sljedovi aktivnosti zasigurno će imati različita trajanja**, ali ono što je ključno je da se sve aktivnosti započinju paralelno i da je potrebno pričekati na završetak svih aktivnosti prije nego se proces prakse može nastaviti.

Primjer:

- **Slijed 1:** (Obavješćavanje poslodavca i evaluacija kandidata kod poslodavca) može trajati nekoliko dana, npr. 3-4 radna dana

- **Slijed 2:** (Studentska priprema za evaluaciju) može trajati nekoliko sati, npr. 2-3 sata ili pak noć prije!
- **Slijed 3:** (Pohrana odabira preferencija u bazu podataka i obavješćavanje studenta o odabranom zadatku) će vjerojatno trajati samo nekoliko sekundi

Ovaj proces možemo detaljnije prikazati raspodjelom u polja i staze:

Definirat ćemo polje PROVOĐENJE STUDENTSKE PRAKSE koje sadrži dvije staze:

- Stazu "Student" budući da nam je student ključni dionik u procesu
- Stazu "Administracija prakse" koja nam predstavlja voditelja, administrativno osoblje i IS kao podlogu.

Poslodavca možemo predstaviti kao vanjskog dionika u procesu kroz zasebno polje POSLODAVAC.

Slika 11: Proces prijave preferencija na studentsku praksu s paralelnim grananjem i spajanjem, prikazan kroz polja i staze

Uočite da smo aktivnost "Evaluacija kandidata" premjestili u polje POSLODAVAC. Poslodavac odrađuje svoj interni proces evaluacije kandidata koji mi apstrahiramo - dovoljno je prikazati kako poslodavac prima obavijest o kandidatu, provodi evaluaciju i završava proces evaluacije.

Nakon paralelnog grananja AND skretnicom, sada nemamo više aktivnost "Evaluacija kandidata" gdje je bilo označeno trajanje.

Kako ćemo sada naglasiti čekanje u Slijedu 1?

► Spoiler alert! Odgovor na pitanje

Slika 12: Proces prijave preferencija na studentsku praksu s paralelnim grananjem i spajanjem, prikazan kroz polja i staze. Uočite prijamni međudogađaj obavijesti koji označava čekanje na rezultat evaluacije poslodavca.






3. Inkluzivna (eng. Inclusive) skretnica

Inkluzivna (OR) skretnica (eng. *Inclusive gateway*) koristi se za modeliranje situacija **baziranih isključivo na podacima** (vrijednostima u procesnoj instanci) gdje se **odabire jedan ili više izlaznih tokova**, odnosno provode se aktivnosti **na svim sljedovima za koji su uvjeti ispunjeni**.

Kao i kod XOR i AND skretnica, i kod inkluzivne (OR) skretnice postoje skretnice **grananja** i **spajanja**.

- Ako inkluzivna skretnica ima više izlaznih tokova, potrebno je definirati **uvjete za svaki izlazni tok**.
- Ako inkluzivna skretnica ima samo jedan tok, onda ne mora imati definiran uvjet.

Inkluzivna skretnica se definira rombom, s oznakom kruga: 

Inkluzivna skretnicu možemo zamisliti kao **logičku disjunkciju** (operator ) , odnosno **odabir jednog ili više uvjeta**. Ukoliko je zadovoljen uvjet, izvršava se odgovarajući slijed aktivnosti. Na neki način radi se **mixu** između  i  skretnice zato što je moguće da se izvrši **samo jedan tok** (kao kod  skretnice) ili **više tokova** (kao kod  skretnice).

Primjer, imamo Bitcoin mjenjačnicu te želimo definirati slijed aktivnosti nakon što korisnik zatraži prodaju određene količine Bitcoina. Primitkom ponude, djelatnik mora poduzeti različite aktivnosti ovisno o cijeni transakcije (vrijednosti Bitcoina koju korisnik želi prodati u eurima):

- svakako moramo obraditi narudžbu za svaki iznos koji je veći od 0 eura
- za iznose veće od 1000 eura, moramo zatražiti verifikaciju osobnih podataka korisnika
- za iznose veće od 10 000 eura, moramo zatražiti odobrenje nadređenog

Zamislimo da su ishodi procesa uvijek isti: "Priprema računa" i "Isplata na bankovni račun korisnika".

U opisanom procesu barem jedan uvjet će uvijek biti zadovoljen (`iznos > 0 eura`).

- mogu biti zadovoljena 2 uvjeta (`iznos > 0 eura` i `iznos > 1000 eura`)
- mogu biti zadovoljena 3 uvjeta (`iznos > 0 eura` , `iznos > 1000 eura` i `iznos > 10 000 eura`)

Slika 11: Proces prodaje Bitcoina u mjenjačnici definiran kroz inkluzivne skretnice grananja i spajanja

Kod **inkluzivnog grananja**, moramo uzeti nekoliko stvari u obzir:

- Iako se može istovremeno ispuniti više uvjeta, moguće je i da ne bude ispunjen nijedan. Međutim, **moramo definirati barem jedan uvjet koji će uvijek biti zadovoljen kako ne bi došlo do *deadlocka* procesa.**
- Ako postoji više uvjeta koji su aktivirani i sljedovi će se izvršiti, inkluzivna skretnica spajanja će pričekati na **sve zadovoljene uvjete koji su "aktivirani"** prije nego nastavi dalje s izvođenjem procesa.

Česta greška, iako na prvu nije očita, jest **ne definiranje uvjeta za svaki izlazni tok i definiranje defaultnog uvjeta**. Defaultni tok je tok koji smo rekli da označavamo oznakom `/` i koji će se izvršiti samo ako nijedan od preostalih uvjeta nije zadovoljen.

Slika 12: Proces prodaje Bitcoina u mjenjačnici s defaultnim tokom (**neispravno zbog defaultne grane**)

Prisjetite se: Zadani (*default flow*) izvršava se onda kada **nijedan od definiranih uvjeta nije zadovoljen**.

Uočite problem:

- ako je zadovoljen uvjet `iznos > 1000 eura`, neće se izvršiti *defaultni* tok i neće se izvršiti aktivnost "Obrada narudžbe"
- ako je zadovoljen uvjet `iznos > 10 000 eura`, neće se izvršiti *defaultni* tok i neće se izvršiti aktivnost "Obrada narudžbe"
- ako je zadovoljen uvjet `iznos > 1000 eura` i `iznos > 10 000 eura`, neće se izvršiti *defaultni* tok i neće se izvršiti aktivnost "Obrada narudžbe".
- u svim drugim slučajevima (`iznosi između 1 i 1000 eura`), izvršit će se samo aktivnost "Obrada narudžbe", što je OK.

VAŽNO: Ista skretnica grananja ne mora "dolaziti u paru" s istom skretnicom spajanja. Npr. `XOR` → `XOR`, `AND` → `AND`, ili `OR` → `OR`. Moguće je (i vrlo učestalo) kombinirati različite skretnice grananja i spajanja ovisno o potrebama modela procesa.

Što bi se dogodilo kad bi za skretnicu spajanja koristili `AND` skretnicu? Kako onda interpretiramo model?

Slika 13: Proces prodaje Bitcoina u mjenjačnici s AND skretnicom spajanja (**neispravno - kriva skretnica spajanja**)

Problem je sljedeći: `AND` skretnica spajanja će pričekati na **sve ulazne tokove** prije nego nastavi dalje, a ne samo na one koji su "aktivirani" (zadovoljeni) kao što to čini `OR` skretnica spajanja.

Primjerice, ako je korisnik prodao Bitcoin u iznosu od 750 eura, skretnica spajanja će čekati na izvršenje sve 3 aktivnosti prije nego nastavi dalje, međutim dvije (druga i treća) neće nikada biti izvršene jer uvjeti nisu zadovoljeni - ovdje dolazi do *deadlocka* procesa.

Primjer posluživanja više vrsta jela u restoranu: Skupina gostiju dolazi u restoran gdje se poslužuje hrana po prethodnoj rezervaciji za veće skupine gostiju. Nakon što se gosti smjeste, konobar donosi jelovnik i gosti biraju jelo. Radi jednostavnosti, recimo da gosti biraju između mesne, vegetarijanske i riblje plate. Međutim, kako ima puno gostiju za stolom, vjerojatno je da će odabrati više različitih vrsta jela. Odabir hrane možemo modelirati kroz inkluzivnu skretnicu.

Slika 14: Proces posluživanja više vrsta jela u restoranu kroz inkluzivnu skretnicu grananja i spajanja

Objasnite kako se ponaša ovaj model?

Na ovom primjeru također možemo kombinirati skretnice grananja i spajanja različitih tipova. Recimo možemo kombinirati skretnicu grananja **OR** sa skretnicom spajanja **XOR**. Kako onda interpretiramo model?

► Spoiler alert! Odgovor na pitanje

Ako postavimo **OR** skretnicu grananja za odabir jela, a naši gosti naruče sve 3 vrste jela, a pritom postoji **XOR** skretnica spajanja, tada će, prema trenutnom modelu, **konobar pričekati na prvu pripremljenu platu i poslužiti će je gostima čim je spremna** - nakon toga proces završava (što nije ono što želimo), budući da se preostala hrana još uvijek priprema.

Ako se naruče sve 3 vrste jela, **OR** skretnica će osigurati "3 ispaljivanja signala", dakle sve što moramo modelirati je petlju gdje konobar čeka na signale o gotovoj hrani jednom kad dostavi prvu platu.

Kako bi proces bio čitljiviji, podijelit ćemo ga u dvije staze "Konobar" i "Kuhinja".

Ne znamo koliko će jela gosti naručiti, niti ne znamo vrijeme pripreme svakog jela, čekanje ćemo definirati kroz **prijamni međudogađaj obavijesti** (eng. *Message Intermediate Catch Event*) koji će konobar "uhvatiti signal" svaki put kad kuhinja javi da je jelo spremno. Nakon toga, konobar poslužuje jelo, a nakon što su gosti pojeli (prikazujemo predajnim međudogađajem - *milestone*), konobar sakuplja prazne tanjure i pribor te **ukoliko se priprema još hrane** (**XOR** skretnica grananja), ponavlja ciklus čekanja na obavijesti ili završava proces.

Slika 15: Proces posluživanja više vrsta jela u restoranu, modeliran na način da konobar dostavlja jela čim su spremna. Primjer pokazuje kako je moguće kombinirati različite vrste skretnica grananja i spajanja.

To je to! Sada smo u jednom modelu prikazali inkluzivnu, paralelnu i ekskluzivnu skretnicu te kombinirali uzastopno različite tipove skretnica grananja i spajanja.

Uočite sljedeće skretnice (s lijeva na desno):

- **Prva paralelna AND skretnica** je skretnica **grananja** (*split/branching*) budući da konobar paralelno kreće s postavljanjem pribora za jelo, a kuhinja kreće s pripremom hrane.
- **Inkluzivna OR skretnica** je skretnica **grananja** budući da su gosti mogli naručiti jednu, dvije ili sve tri vrste jela.
- **Ekskluzivna XOR skretnica** je skretnica **spajanja** (*join/merge*) budući da se svaka plata poslužuje čim je spremna (proces nastavlja čim stigne jedan signal - međutim svi pokrenuti će kad tad stići)
- **Druga inkluzivna OR skretnica** je skretnica **spajanja** koja služi za implementaciju petlje; ako se dostavlja prvo jelo, skretnica "propušta" odmah jer je 1/1 signal stigao, ali ako se naručilo više jela, te

jedno već dostavilo, skretnica će čekati na sljedeći signal (sljedeće jelo koje je spremno) i tako dalje dok se ne dostave sva jela. Potencijalno bi se ovdje mogla koristiti i XOR varijanta, međutim OR je prikladnija.

- **Druga ekskluzivna ~~xor~~ skretnica** je skretnica **grananja** koja odlučuje hoće li se proces ponoviti (ako se priprema još hrane) ili će završiti (ako su svi gosti pojeli sve naručeno).

4. Ukratko, kada koristiti koju skretnicu?

Do sad smo prošli kroz tri vrste skretnica u BPMN-u (premda ih ima još nekoliko, ove su najčešće korištene):

1. **Ekskluzivna (XOR) skretnica**
2. **Paralelna (AND) skretnica**
3. **Inkluzivna (OR) skretnica**

Kada koristiti koju skretnicu (u jednoj rečenici):

1. **XOR skretnica** koristi se za odabir jedne opcije između više opcija, gdje se uvijek **odabire samo jedna opcija** za koju je **predikat/uvjet zadovoljen**.
2. **AND skretnica** koristi se za modeliranje situacija u kojima se više aktivnosti izvršava **paralelno**, a zatim se nastavlja dalje tek kada su **sve aktivnosti završene**.
3. **OR skretnica** koristi se za odabir jedne ili više opcija između više opcija, gdje se **odabiru sve opcije** za koje je **definirani logički izraz istinit**.

Također, vidjeli smo da za svaku skretnicu možemo definirati skretnicu **grananja** (eng. *split*) i skretnicu **spajanja** (eng. *join/merge*) kako bismo preciznije definirali tokove u procesu.

1. **XOR skretnica spajanja**: koristi se kada postoji više od dvije moguće grane toka te **omogućuje nastavak procesa nakon završetka prvog završenog slijeda**. Ako postoje samo dvije opcije, skretnicu nije nužno eksplicitno definirati (ali može se i uobičajeno je).
2. **AND skretnica spajanja**: koristi se za **sinkronizaciju svih paralelnih aktivnosti**, odnosno nastavlja proces tek nakon što su svi sljedovi aktivnosti pokrenuti paralelnim grananjem završeni.
3. **OR skretnica spajanja**: koristi se kada je potrebno **čekati završetak samo onih aktivnosti koje su zadovoljile određeni uvjet**. Dakle, proces se nastavlja nakon završetka svih relevantnih, ali ne nužno svih, aktivnosti. **Proces neće započeti završetkom prve aktivnosti** (osim ako nije jedina zadovoljila uvjet), već će pričekati na sve aktivnosti koje su "aktivirane" (zadovoljeni uvjeti).

Važno je istaknuti da kod skretnica spajanja nije nužno da svi ulazni tokovi potječu iz iste skretnice grananja. Moguće je da se više skretnica grananja povezuje u jednu skretnicu spajanja, što omogućuje veću fleksibilnost pri modeliranju složenijih procesa. Takav pristup prikazan je u posljednjem primjeru s restoranom, na Slici 13.

Zadaci za Vježbu 3

1. Wolt - dostava hrane

Modelirajte proces naručivanja hrane putem Wolt aplikacije. Proces započinje onoga trenutka kada u restoran pristigne narudžba s Wolt aplikacije. Nakon što djelatnik obradi narudžbu, paralelno započinje priprema hrane i obavješćavanje dostavljača. Dostavljač, kao samostalni vanjski sudionik, nije dio organizacije restorana te sudjeluje samo u procesu dostave hrane. Taj proces započinje kada dostavljaču pristigne obavijest o traženoj dostavi. Dostavljač pregledava obavijest i odlučuje hoće li prihvatiti dostavu. Ako odbije, njegov proces tu završava i o tome obavještava restoran. Ako prihvati, obavještava restoran da će preuzeti dostavu.

Međutim, dostava se može preuzeti tek nakon što je hrana pripremljena i zapakirana, što zahtijeva određeno vrijeme. U tom razdoblju restoran čeka potvrdu od dostavljača da će zaista preuzeti i dostaviti naručenu hranu. Ako je povratna informacija pozitivna (dostavljač prihvaća dostavu unutar otprilike 20 minuta), a hrana je zapakirana (spremna za preuzimanje), tada se slijed može nastaviti. Ako je povratna informacija negativna, u sustavu se zatraži novi dostavljač i čeka se njegova potvrda. Kada su svi uvjeti zadovoljeni, korisnik se obavještava da je hrana na putu te se istovremeno izrađuje račun i obavještava dostavljača da je hrana spremna za preuzimanje. Kada dostavljač primi tu informaciju, preuzima hranu u restoranu, dostavlja je korisniku i time proces završava.

2. Liječenje pacijenta koji dolazi na hitnu pomoć

Liječenje pacijenta u bolnici koji dolazi na hitnu pomoć započinje njegovim dolaskom i prijavom na šalteru hitnog prijema. Medicinski tehničar unosi osnovne podatke o pacijentu u informacijski sustav te obavještava dežurnog liječnika. Nakon toga slijedi početna trijaža u kojoj medicinska sestra procjenjuje težinu stanja pacijenta i određuje razinu hitnosti. Ako je stanje pacijenta životno ugrožavajuće, odmah se upućuje u sobu za hitne intervencije, gdje paralelno započinju mjerenje vitalnih funkcija i pozivanje dežurnog liječnika specijalista. Ako pacijent nije u kritičnom stanju, čeka pregled u čekaonici dok se ne oslobodi liječnik; ako čeka više od 1 sat, medicinska sestra ponovno procjenjuje njegovo stanje i, ako je potrebno, ažurira razinu hitnosti, a ako čeka više od 3 sata, pacijent shvaća da mu ipak nije toliko loše i odlučuje otići kući.

Liječnik zatim pregledava pacijenta i odlučuje o daljnjim postupcima. Ako je potrebno, izdaje naloge za laboratorijske pretrage i dijagnostičke preglede (rendgen, CT, ultrazvuk). Pacijent odrađuje dijagnostičke pretrage jednu po jednu, ali ne nužno definiranim redoslijedom, već kako se koji uređaj oslobodi. Nakon što rezultati stignu, sustav ih automatski pridružuje pacijentovu kartonu i obavještava liječnika da su dostupni. Na temelju rezultata liječnik donosi odluku o liječenju. Ako je potrebno bolničko liječenje, pacijent se upućuje na odjel, a ako nije, propisuje se terapija i pacijent se otpušta kući uz upute o daljnjem postupanju. U slučaju da tijekom obrade dođe do pogoršanja stanja, proces se vraća na hitnu intervenciju. Proces završava kada pacijent napusti hitnu službu ili bude službeno primljen na bolnički odjel.

3. Organizacija koncerta u Pulskoj Areni

Proces organizacije velikog koncerta u Pulskoj Areni započinje donošenjem odluke o održavanju događaja i izborom glavnog izvođača. Organizator (npr. produkcijska tvrtka AdriaSound Events) kontaktira menadžment izvođača - primjerice *Coldplay*, *Dua Lipa* ili *Imagine Dragons* — radi pregovora o uvjetima nastupa. Nakon prihvaćanja ponude, paralelno se pokreću tri ključna toka aktivnosti: pravna priprema ugovora, rezervacija prostora te izrada preliminarog proračuna.

Po potpisivanju ugovora započinje priprema koncerta. Tri tima djeluju paralelno i koordinirano:

1. **Organizacijski tim** vodi komunikaciju s izvođačem, dobavljačima i gradskim službama te prati realizaciju ugovorenih obveza.
2. **Produkcijski tim** izrađuje tehničku dokumentaciju, plan pozornice, rasvjete i ozvučenja te koordinira dostavu i postavljanje opreme.
3. **Marketinški tim** osmišljava strategiju oglašavanja i pokreće prodaju ulaznica putem platformi poput Eventim.hr ili Entrio.

Sustav za prodaju ulaznica kontinuirano prati broj prodanih karata. Ako se prodaja odvija sporije od planiranog, marketinški tim pokreće dodatnu kampanju i aktivira sponzorske suradnje. Ako su ulaznice rasprodane, prodaja se automatski zatvara i svi kanali obavještavaju publiku o rasprodanim ulaznicama.

Tijekom završne faze pripreme, produkcijski tim organizira tehničku probu izvođača. Ako se pojave problemi sa zvukom ili rasvjetom, proces se vraća na prilagodbu tehničke konfiguracije dok se ne postigne željena razina kvalitete - ukoliko isto nije moguće, kreće se u hitni pronalazak zamjenske opreme u suradnji s lokalnim dobavljačima.

Na dan koncerta tri tima ponovno djeluju paralelno: organizacijski tim nadzire dolazak izvođača i publike te koordinira sigurnost događaja u suradnji s policijom, vatrogascima i zaštitarskim službama (ulasci ljudi, sigurnosne provjere torbi i sl.), produkcijski tim ponovno provjerava zvuk, svjetlo i specijalne efekte, a marketinški tim koordinira održavanje koncerta s lokalnim medijima i sponzorima. U slučaju *last-minute* nepovoljnih vremenskih uvjeta, meteorološki sustav automatski obavještava organizacijski tim, koji donosi odluku o pomicanju rasporeda ili odgodi koncerta za drugi dan.

Jednom kad koncert započne, izvođač preuzima glavnu pozornicu i započinje program. Tijekom nastupa tehnički tim kontinuirano nadzire zvuk, svjetlo i specijalne efekte, sigurnosni tim nadzire sigurnost publike a marketinški tim na terenu prodaje rekvizite, *merchandise*, hranu i piće. **Nakon završetka koncerta** započinju aktivnosti demontaže, prikupljanja opreme i čišćenja Arene, dok marketinški tim i organizacijski tim prikupljaju povratne informacije od publike i sponzora te izrađuju završno financijsko izvješće. Proces završava kada su svi ugovori i računi zatvoreni, prostor vraćen u prvobitno stanje i završno izvješće poslano gradu Puli i sponzorima.

#Ne postoji idealno rješenje za modeliranje ovih procesa - pokušajte na temelju pomoćnog teksta prikazati što realniju sliku procesa.