

Servisno orijentisana arhitektura

i integracija poslovnih
aplikacija

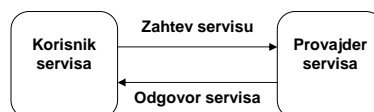
Šta je SOA ?

- **Definicije**

- Oblik organizacije integrisanog informacionog okruženja jednog poslovnog sistema, koji karakteriše ponuda i korišćenje njegovih distribuiranih funkcija, predstavljenih servisima
- “oblik višeslojne organizacije računarskih sistema koji obezbeđuje deljenje logike poslovanja i informacija od strane različitih softverskih sistema i načina njihovog korišćenja”, *Gartner*

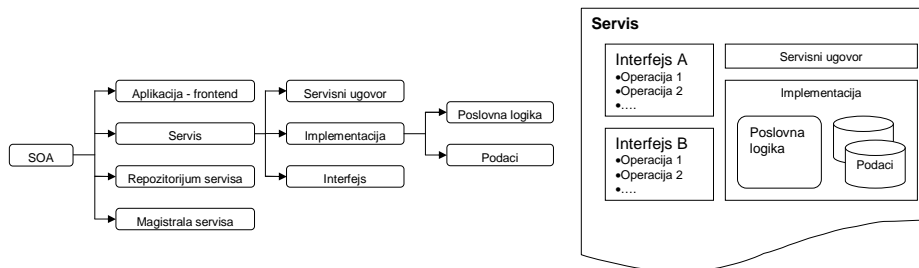
- **Karakteristike**

- granulacija poslovnih funkcija i
- izgradnja arhitekture poslovnog sistema, čija je realizacija nezavisna od tehnologije njene implementacije.
 - neutralna sa stanovišta tehnoloških standarda i protokola
- modeliranje i implementacija poslovne, a ne tehničke infrastrukture



Osnovni elementi SOA

- Aplikacije
 - Uloga je iniciranje poslovnih procesa i prijem rezultata neke aktivnosti servisa. Životni vek aplikacije je odvojen od životnog veka servisa.
- Servis
- Repozitorijum
 - Koristi se za skladištenje servisnih ugovora individualnih servisa
- Magistrala



Principi projektovanja SOA

- **Enkapsulacija servisa.** Servis se predstavlja svojom definicijom, odnosno, interfejsom prema okruženju, koji otkriva njegovu funkciju i odgovarajućom, enkapsuliranom logikom za njeno izvršavanje.
- **Slaba međusobna povezanost.** Iako između servisa postoje relacije, oni ne smeju biti međusobno uslovljeni. Jedina vrsta veze koja se između njih uspostavlja je znanje o međusobnom postojanju.
- **Ekspozicija funkcionalnosti servisa putem servisnog ugovora (Service Contract).** Funkcija i opis servisa se predstavlja servisnim ugovorom (Service Contract).
- **Apstrakcija servisa.** Okruženju su dostupne samo informacije o servisu koje su predstavljene u njegovom ugovoru. Logika funkcionisanja je sakrivena.
- **Višestruko korišćenje servisa.** Logika se distribuira po servisima sa ciljem da se omogući njihovo višestruko korišćenje u različitim kontekstima.
- **Sposobnost kompozicije servisa.** Raznovrsne grupe servisa se mogu orkestrirati i koordinisati sa ciljem formiranja kompozitnih servisa.
- **Autonomija servisa.** Isključivu kontrolu nad logikom koja je enkapsulirana u okviru određenog servisa ima samo taj servis.
- **Neperzistentnost servisa.** Servisi se staraju o očuvanju minimalnog skupa informacija o aktivnostima koje se sprovode.
- **Eksponiranost servisa.** Servisi se projektuju tako da se mogu lako pronaći i analizirati od strane odgovarajućih mehanizama za njihovo otkrivanje.

Aktivnosti razvoja SOA

- Dekompozicija
 - identifikacija hijerarhije servisa, u kontekstu organizacije, procesa i funkcija poslovnog sistema,
 - projektovanje semantičkog modela koji treba da ustanovi smernice za međusobnu komunikaciju servisa i obezbeđivanje njihove interoperabilnosti; i
 - refaktorisanje identifikovanih servisa, u cilju obezbeđenja principa i standarda performansi, proširivosti, bezbednosti, itd.
- Definicija
 - Sadrži informacije na osnovu kojih korisnik treba da utvrdi koji servis mu je potreban. One obuhvataju opis namene i ciljeva servisa, ograničenja u korišćenju i nivo kvaliteta, uslovi koje korisnik treba da ispuni da bi mogao da ga koristi. Pored toga, korisnik treba da zna kako se servis koristi. U tom kontekstu, definicija servisa treba da sadrži strukturu zahteva i isporučenog rezultata servisa, uslove pod kojima dolazi do određenih rezultata, i sl.
 - Sadrži tehničke informacije o pozivanju servisa. Komunikacioni protokoli, formati poruka, uključujući i tehnike serijalizacije, lokacija servisa, zahtevi bezbednosti, opis SOAP zaglavlja, kvantitativno izražene parametre kvaliteta servisa, kao što su vreme dostupnosti, vreme odgovora, obradna moć servisa, itd.
- Implementacija
 - izgradnja posebnog sloja servisa kojim se racionalizuju funkcije postojećih aplikacija

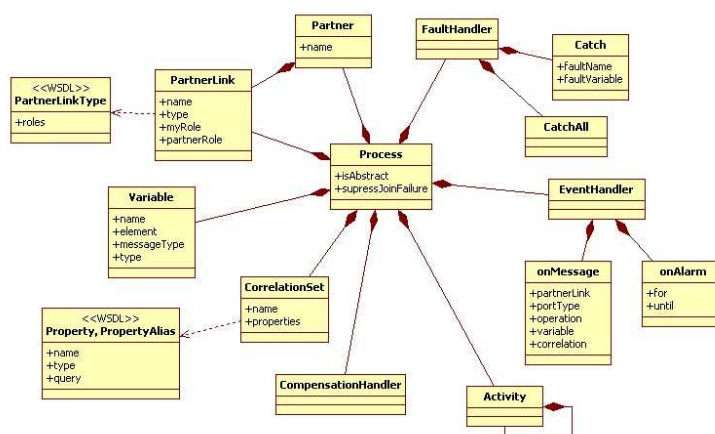
Osnovne karakteristike implementacije poslovnih procesa kao sredstva za orkestraciju servisa u SOA

- Servise karakteriše neperzistentnost, odnosno izuzeće funkcije beleženja trenutnog stanja servisa. S obzirom na to, implementacija poslovnih procesa podrazumeva upravljanje kontekstom izvršenja servisa, odnosno njihovog pozivanja u određenom redosledu, upravljanje tokovima podataka i eventualno skladištenje podataka o stanju procesa.
- Skoro sve poslovne procese karakteriše određeni nivo manualnih aktivnosti. Da bi one mogle da se sprovode na odgovarajući način, potrebno je da implementacija poslovnih procesa obuhvati i podršku radnim listama, pristup sistemu zasnovan na pravima (*role-based*), "zaključavanje" aktivnosti koje se trenutno izvršavaju, itd.
- Implementacija poslovnih procesa kao sredstva za orkestraciju servisa predstavlja autonoman element servisno orijentisane arhitekture, potpuno nezavisan od tehnologije razvoja servisa, odnosno infrastrukture korišćene za njihovu koordinaciju.
- Poslovni procesi predstavljaju sloj realizacije servisno orijentisane arhitekture koji je, za razliku od implementacije samih servisa, podložan kratkoročnim promenama. U tom smislu, neophodno je da oni budu realizovan na fleksibilan, modularni, proširivi način, primenom odgovarajućih široko usvojenih standarda.
- Osnovni elementi orkestracije servisu su *engine* za orkestraciju i jezik za orkestraciju. *Engine* predstavlja aplikacioni softver (server) koji omogućava izvršavanje jednog ili više jezika za orkestraciju.
- Jezik za orkestraciju servisa predstavlja skup semantičkih i sintakasnih pravila za formulisanje i kodiranje strukture koordinacije servisa.

BPEL jezik za modeliranje procesa

- BPEL4WS model procesa predstavlja objedinjenu specifikaciju web servisa (WSDL – *Web Services Definition Language*) i sintakse za njihovu orkestraciju u poslovni proces.
- Ljudska interakcija sa procesom nije obuhvaćena BPEL specifikacijom. Ona je, pre svega, namenjena podršci izvršenju automatskih poslovnih procesa, zasnovanih na orkestriranju postojećih web servisa – poslovnih funkcija servisno orijentisane arhitekture

Elementi BPEL specifikacije 1/2



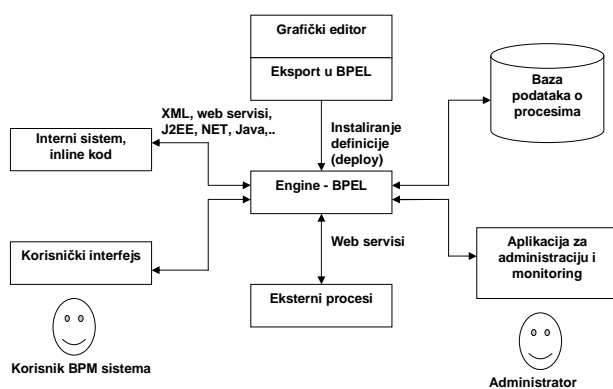
Elementi BPEL specifikacije 2/2

Naziv objekta	Mesto definicije	Opis
<i>Process</i>	BPEL	Poslovni proces koji sadrži jedan ili više drugih objekata.
<i>Variable</i>	BPEL	Promenljiva koju koristi proces – tip zasnovan na WSDL tipu poruke, XSD element ili osnovni XSD tip. Proces može imati nijednu ili više promenljivih. Sve promenljive čije se vrednosti razmenjuju u okviru interakcije jednog procesa sa partnerskim servisom se moraju deklarirati u okviru procesa. Dodatno, njihove vrednosti se mogu koristiti za jednostavnu obradu u okviru procesa.
<i>Property, Property Alias</i>	WSDL	Osobina predstavlja jedan token podataka WSDL poruke. Alias osobine je <i>XPath</i> izraz koji se koristi za pronalaženje vrednosti osobine. Osobine se koriste za definisanje korelacije BPEL promenljivih.
<i>CorrelationSet</i>	BPEL	Skup korelacija predstavlja skup jedne ili više osobina koje se koriste za uzajamno referenciranje podataka iz jedne poruke i stanja procesa.
<i>Partner Link Type</i>	WSDL	Tip partnerskog linka predstavlja definiciju referenciranja tipova portova na partnerske role.
<i>Partner Link</i>	BPEL	Partnerski link predstavlja deklaraciju procesa o tome koje veze sa partnerskim web servisima su podržane.
<i>Partner</i>	BPEL	Skup partnerskih linkova
<i>Compensation Handler</i>	BPEL	Aktivnost koja sadrži logiku otkazivanja ili povraćaja procesa na prethodno stanje. Ona se izvršava u slučaju da je proces koji je već izvršen potrebno vratiti u inicijalno stanje.
<i>Fault Handler, Catch, CatchAll</i>	BPEL	Skup handlera za obradu izuzetaka, na osnovu tipa greške, unutar jednog procesa. Obrada izuzetaka, ili grešaka, se veoma često vrši prilikom sinhronog poziva operacije partnerskog web servisa. Prihvat greške (<i>catch</i>) se može vršiti u okviru aktivnosti procesa, prilikom čega ona poziva drugu, namensku aktivnost koja vrši obradu greške. Ukoliko aktivnost procesa ne predviđa prihvat greške, on se vrši u okviru odgovarajućeg okvira (<i>scope</i>) - autonomnog dela procesa sa svojim skupom handlera, promenljivih i skupova korelacija.
<i>EventHandler, onMessage, onAlarm</i>	BPEL	Skup handlera za obradu događaja unutar jednog procesa.
<i>Activity</i>	BPEL	Osnovni tip BPEL aktivnosti. BPEL specifikacija definiše da se proces sastoji od samo jedne aktivnosti koja je strukturirana u posebne delove.

Tipovi BPEL aktivnosti

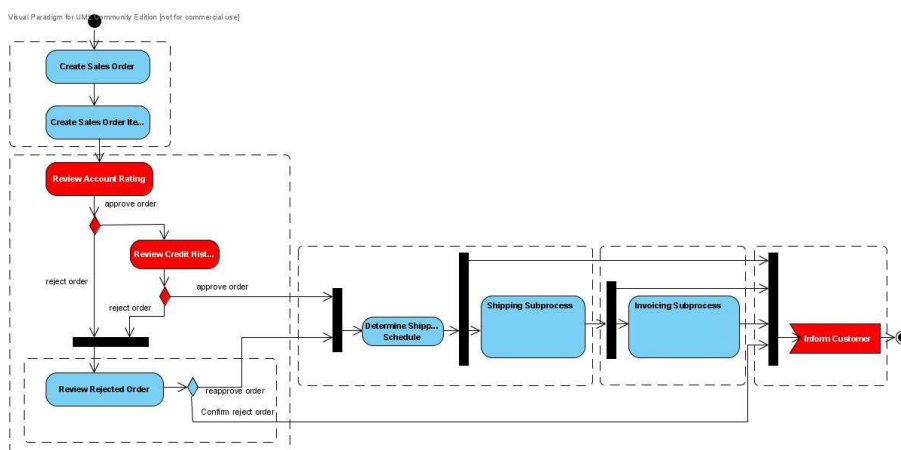
Tip aktivnosti	Opis
<i>Receive</i>	Aktivnost koja prima SOAP poruku od strane nekog web servisa. Poslovni proces obezbeđuje pristup web servisima svojim partnerima kroz aktivnosti prijema poruka (<i>receive</i>) i odgovarajućih odgovora (<i>reply</i> aktivnosti). U okviru <i>receive</i> aktivnosti, definiše se link partnera od kojeg se očekuje slanje poruke procesu, odnosno očekivani tip porta i operacija web servisa. Ovaj tip aktivnosti predstavlja inicijalnu aktivnost svakog procesa, pri čemu, više uporednih <i>receive</i> aktivnosti mogu predstavljati ulaz u proces – jezgro njegovog instanciranja, pod uslovom da je njihov parametar <i>createInstance</i> , podešen na <i>yes</i> . S obzirom na to da <i>receive</i> aktivnost predstavlja obradu prijema neke poruke, očigledno je da ona "blokira" proces – proces se ne može instancirati, odnosno, nastaviti, sve dok se ne izvrši prijem poruke.
<i>Reply</i>	Aktivnost koja vraća sinhroni odgovor web servisu, nakon prijema odgovarajuće poruke (<i>Receive</i> aktivnosti).
<i>Invoke</i>	Aktivnost koja poziva partnerski web servis, na sinhroni ili asinhroni način. Specifikacija asinhronog poziva obuhvata definisanje ulazne promenljive – parametra zahteva, dok se odziv ne definiše. Kod sinhronog poziva, definiše se parametri - promenljive zahteva i odziva. Tom prilikom je potrebno, ali ne i neophodno, obezbediti prihvat i obradu greške (<i>fault handling</i>), koji može da generiše WSDL interfejs.
<i>Compensate</i>	Aktivnost koja aktivira kompenzaciju procesa.
<i>Throw</i>	Aktivnost koja registruje grešku u odvijanju procesa i aktivira izvršavanje handlera grešaka u procesu.
<i>Assign</i>	Aktivnost koja kopira vrednost jedne promenljive u drugu, odnosno vrši manipulaciju jednom promenljivom, tako što je primenom <i>XPath</i> , <i>XQuery</i> ili <i>XSLT</i> upita, izraza ili funkcija transformiše, a rezultat transformacije dodeljuje drugoj promenljivoj.
<i>Wait</i>	Aktivnost koja zaustavlja izvršavanje procesa u definisanom trajanju, ili do zadatog vremena.
<i>Empty</i>	Prazna aktivnost. Ne izvršava ništa.
<i>Switch</i>	Ekskluzivna OR struktura. Izvršava aktivnost samo u slučaju da je vrednost određenog izraza - <i>true</i> .
<i>Flow</i>	Struktura sa različitim paralelnim aktivnostima. Sve aktivnosti koje su definisane unutar <i>Flow</i> bloka se izvršavaju uporedno.
<i>Sequence</i>	Sekvencijalno izvršavanje skupa aktivnosti. Sve aktivnosti koje su definisane unutar <i>Sequence</i> bloka se izvršavaju postavljenim redosledom.
<i>Pick</i>	Struktura koja čeka na događaje. Izvršava aktivnost u njenom domenu, za koju je definisano da se izvršava prilikom određenog događaja (<i>onMessage</i> , <i>onEvent</i> , <i>onAlarm</i>).
<i>While</i>	Izvršavanje aktivnosti u petlji za svo vreme za koje je vrednost zadatog <i>XPath</i> izraza – <i>true</i> .
<i>Scope</i>	Poseban autonomni deo procesa sa svojim skupom handlera, promenljivih i skupova korelacija.

Integrirani BPM sistemi

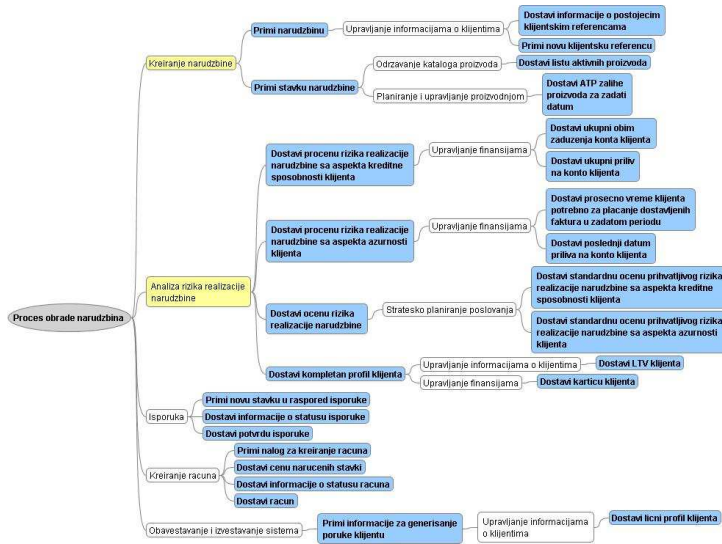


- grafičko modeliranje procesa,
- *engine* za izvršavanje definisanih poslovnih procesa,
- alati i interfejsi za interakciju korisnika i drugih podsistema sa BPM sistemom,
- administracija i monitoring poslovnih procesa,
- jedinstveni model podataka o procesima i
- upravljanje verzijama procesa

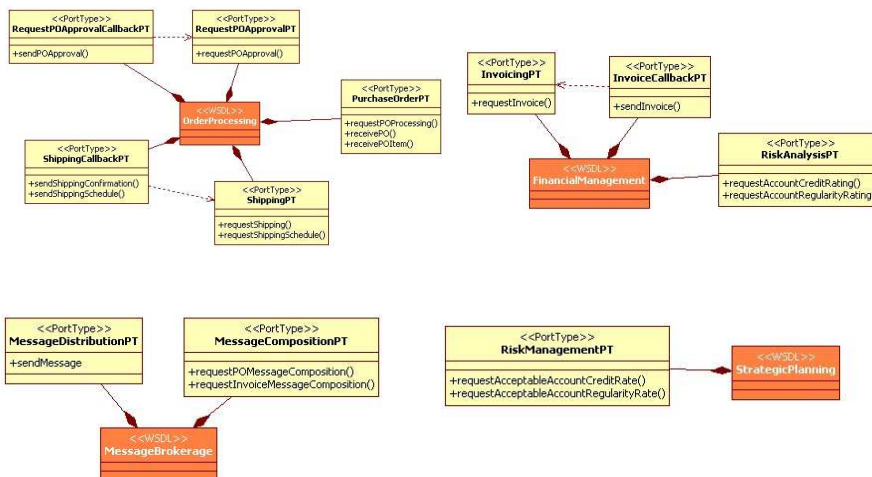
Primer procesa obrade narudžbina



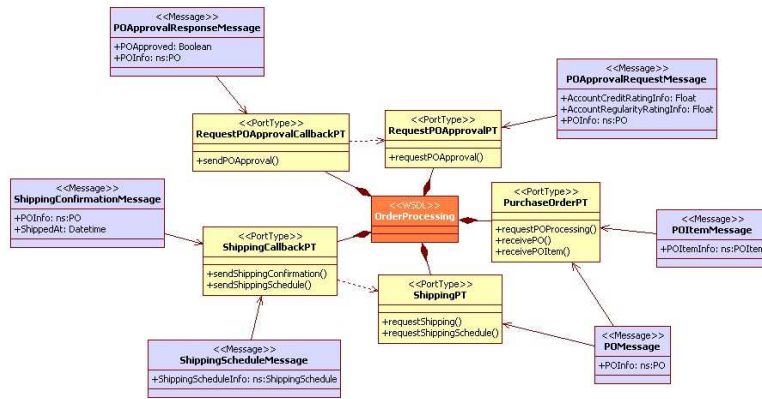
Dekmpozicija procesa obrade narudžbina



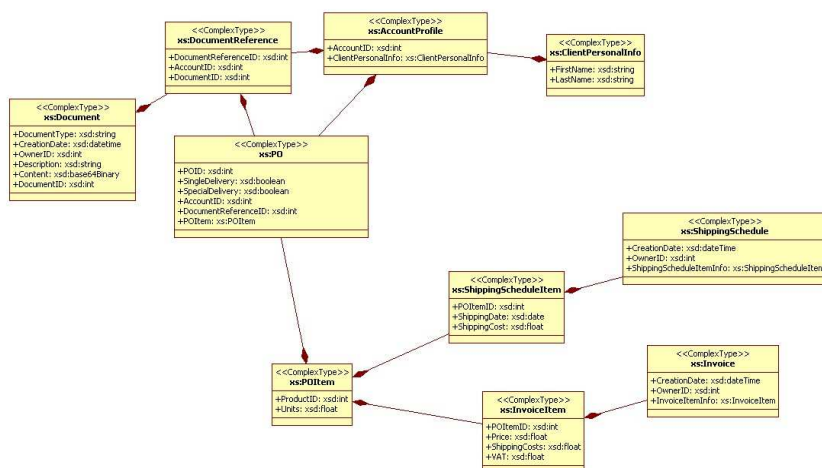
Modeli interfejsa



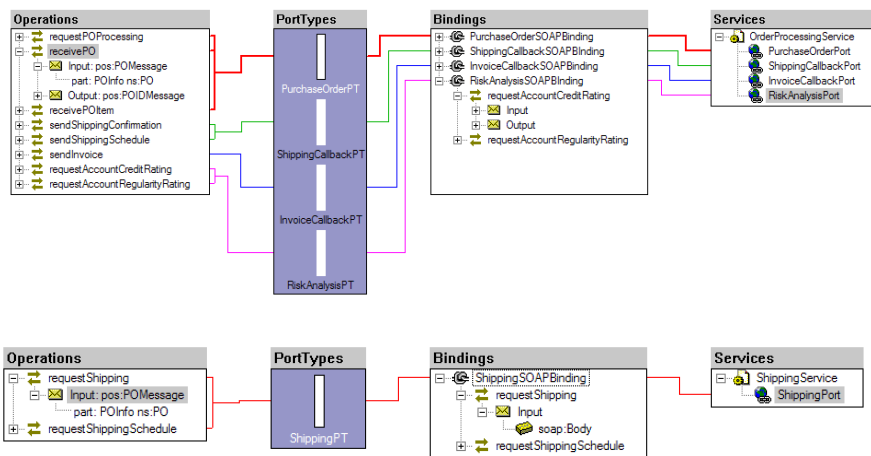
Semantički modeli interoperabilnosti partnera



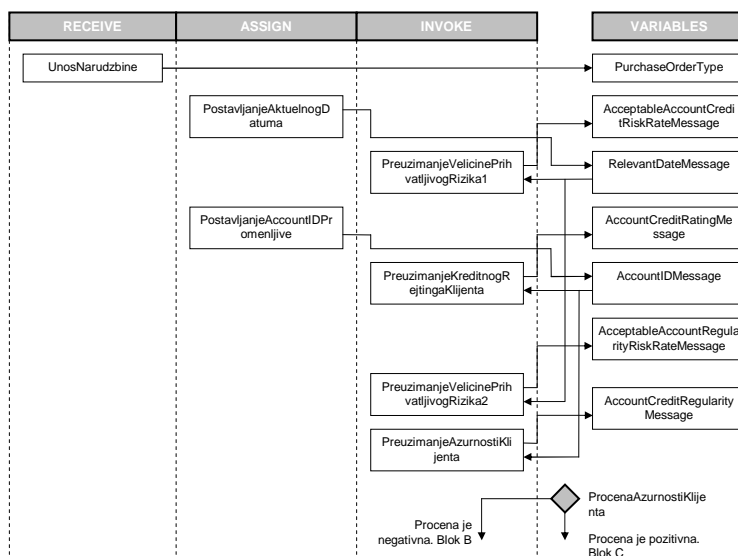
Ukupni model podataka - XSD



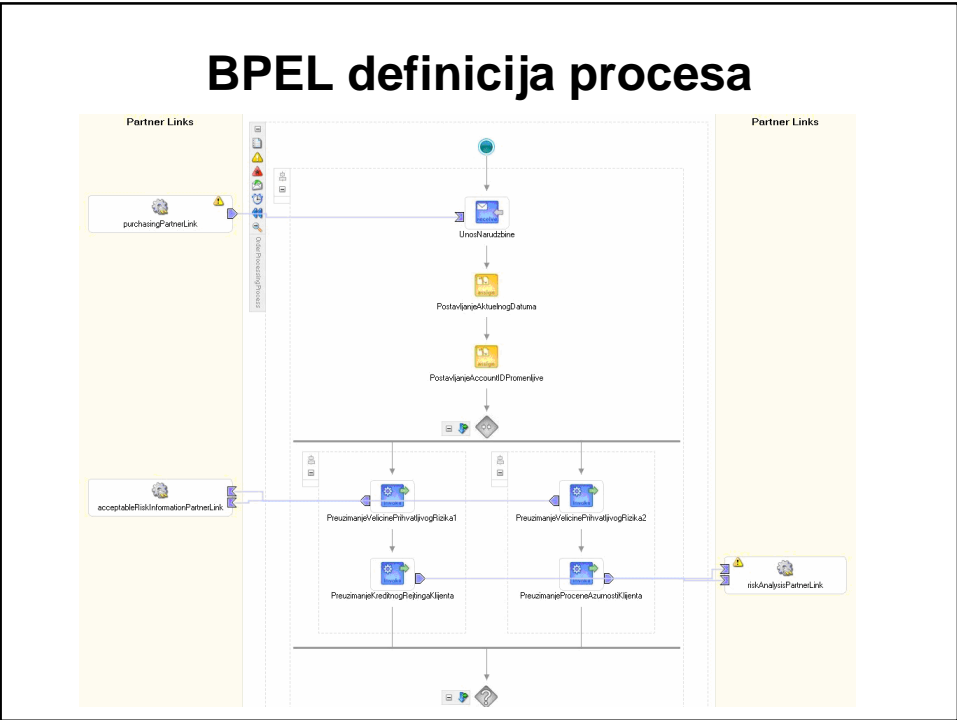
WSDL Interfejsi



Apstraktna definicija procesa



BPEL definicija procesa



BPEL definicija procesa

