

Osnovni koncepti

UPP koriste dve vrste ljudi, menadžeri i IT stručnjaci. Menadžeri koriste upp za poboljšavanje sposobnosti kompanije (veće zadovoljstvo krajnjih korisnika, poboljšavanje proizvoda, smanjivanje troškova..). Upravljanje poslovnim procesima je zasnovano na postavci da je svaki proizvod ili usluga koju određena kompanija nudi rezultat obavljanja niza aktivnosti u kompaniji. IT sektor pomaže da se realizuju te aktivnosti kroz informacione sisteme. Redoslijed aktivnosti u poslovnom procesu je bitan. Poslovni proces se izvršava u okviru jedne organizacije ali može imati interakcije sa poslovnim procesima drugih organizacija. Sistem za upravljanje poslovnim procesima je generički softver koji obezbjeđuje koordinisano izvršavanje poslovnog procesa na osnovu eksplicitne reprezentacije poslovnog procesa. Eksplicitna reprezentacija poslovnog procesa može biti tekstualna (nedostaje struktuiranost reprezentacije) i formalna (model poslovnog procesa – dijagrami). Sistem za upravljanje poslovnim procesima se koristi za uspostavljanje kontrole nad poslovnim procesom. Upravljanje poslovnim procesom se drugačije naziva "**orkestracija**" procesa. **Koreografija** procesa predstavlja interakciju poslovnih procesa različitih organizacija. Životni ciklus poslovnog procesa se sastoji od četiri faze:

- Dizajn i analiza

Dizajn i analiza obuhvata modelovanje kao ključnu aktivnost, validaciju i simulaciju. Modelovanje je evolutivnog karaktera.

- Konfigurisanje

Konfigurisanje se svodi na izbor platforme za implementaciju, implementaciju i testiranja. Vrste testova su:

- Testovi funkcionalnosti
 - Integracioni testovi
 - Testovi performansi
- Izvršavanje

Pokreću se instance poslovnog procesa da bi se ostvarili ciljevi kompanije. Sistem za upravljanje poslovnim procesima vrši aktivnu kontrolu nad izvršavanjem poslovnog procesa na osnovu definisanog modela. Prilikom izvršavanja mogu da se vide trenutne aktivnosti u procesima, pokrenuti procesi i završeni procesi.

- Evaluacija

Ocjena poslovnog procesa na osnovu iskustva stečenog tokom korišćenja sistema i podataka prikupljenih tokom izvršavanja poslovnog procesa koji mogu da se nalaze u log file-ovima ili da koristimo *process mining*. Zaključci evaluacije se koriste za unapređenje modela.

Tipovi učesnika u poslovnim procesima:

- Glavni menadžer poslovnih procesa
- Inženjer poslovnih procesa (Eksperti u nekoj poslovnoj oblasti)
- Dizajner poslovnih procesa (Zaduženi za modelovanje poslovnih procesa)
- Učesnici u izvršenju procesa

- *Obučeni radnici*
- *Vlasnik poslovnog procesa*
- *Arhitekta sistema* (Zadužen za razvoj softverskog sistema za upp)
- *Programeri* (Zaduženi za razvoj komponenti softverskog sistema, interfejsi)

Poslovni procesi mogu da se podijele:

- Organizacione/operativne
- Interne/koreografije
- Po stepenu automatizacije (potpuno automatizovani, sistemi koji zahtjevaju aktivnosti korisnika i manualni sistemi)
- Po stepenu ponovljivosti (često i rijetko ponovljivi)
- Po stepenu struktuiranosti (produkcioni, adhoc)

Organizacioni/operativni imaju četiri nivoa apstrakcije, najviši koji se odnosi na poslovne ciljeve i strategije, drugi nivo koji se tiče organizacije, treći nivo se bavi operativnim poslovnim procesima i četvrti je implementacioni nivo.

Ciljevi upravljanja poslovnim procesima su bolje razumijevanje funkcionisanja organizacija putem eksplicitne reprezentacije poslovnih procesa, mogućnost lakše adaptacije na promjene poslovnog procesa, evolutivni razvoj, i smanjenje jaza između poslovnih procesa i softverskih sistema koji pružaju podršku njihovom izvršavanju.

Evolucija poslovnih sistema

Poslovni sistemi se sastoje od informacionih sistema. Softver se organizuje u module koji omogućavaju ponovnu iskoristivost modula, njihovu lakšu modifikaciju, sakrivanje informacija (lokalne promjene ne izazivaju promjene cijelog sistema). Razlozi zbog kojih može doći do promjene softvera su novi zahtjevi korisnika, tehnološke promjene, promjene zakonskih okvira... Kod sistema za upravljanje poslovnim procesima, prilagodljivost promjenama je jedan od glavnih ciljeva.

Enterprise resource planning (ERP) sistemi su razvijeni sa namjerom da objedine fragmentirane podatke u jednu bazu podataka. Umjesto više aplikacija, bilo je više modula ERP-a. Tu su još *Supply chain management* (SCM) i *Customer relationship management* (CRM). Uvođenje specijalizovanih sistema do koncepta "*silos aplikacija*", koje imaju skoro nikakvu integraciju. Javlja se problem heterogenosti podataka (isti podaci različito definisani u različitim aplikacijama). Zbog toga se uvodi srednji sloj koji služi za integraciju poslovnih aplikacija.

Point-to-point integracija direktno povezuje svaki par aplikacija, problem predstavlja N^2 interfejsa.

Srednji sloj je baziran na razmjeni poruka, srednji sloj samo kontroliše i garantuje isporuku *Hub-and-spoke* integracija se bazira na centralnom čvorištu i aplikacijama koje su na njega spojene bez direktne veze između pojedinih aplikacija. I ovaj princip funkcionise na razmjeni poruka, ali one više ne sadrže eksplicitno informaciju ko je primalac nego *hub* na osnovu sadržaja i strukture određuje kojoj aplikaciji da isporuči poruku. Broj veza se redukuje na N . *Hub* određuje adaptere preko kojih se ostvaruje veza sa aplikacijama. Srednji sloj se može realizovati i pomoću "message brokera" (korisnik definiše pravila i format komunikacije, implementacija adaptera se svodi na deklaraciju odgovarajućih pravila, koristi se *publish/subscribe* mehanizam).

//Boze sacuvaj

Upravljanje radnim procesima (workflow management) i upravljanje poslovnim procesima zasnovano je na konceptima vrijednosnih lanaca (value-chains) i orijentacije na procese.

Vrijednosni lanci (value chains) prikazuju poslovne funkcije visokog nivoa bez prejeranih detalja, opisuju globalni tok informacija i dobara, poslovne funkcije prikazane u strukturi lanaca.

Organizacioni poslovni procesi opisuju poslovne procese na visokom nivou (organizacionom nivou).

Business to business (B2B) opisuju interakciju na visokom nivou (prikazuju veze vrijednosnih lanaca različitih kompanija). Problemi pri modelovanju ovakvih procesa su deadlocks, javno prikazivanje internih poslovnih procesa.

Radni proces (Workflow) predstavlja automatizaciju poslovnog procesa. Postoje single i multiple application workflow/s. Postoje sistemski i interaktivni radni procesi. Kod sistemskih radnih procesa aktivnosti se izvršavaju automatski. Interaktivni radni procesi zahtjevaju interakciju sa korisnikom. Tipično se koriste kao dio većeg poslovnog procesa koji ima i automatizovane podprocese. Za svaku aktivnost se definiše koji korisnici je mogu izvršiti (role).

Problemi ovih sistema su : Pitanje raspodjele zadataka može biti opterećavajuće, mora se obezbijediti fleksibilnost ekspertskim korisnicima pazeći da se ne naruši opšti tok procesa. Aplikativni sistemi koje treba integrisati sa workflow sistemom imaju nedovoljno dokumentovane interfejs, povezanost aplikacija na niskom nivou je realizovano na nivou programskog koda i kao takvo, teško je preslikati u procesni model.

Modeliranje poslovnih procesa

Modelovanje poslovnih procesa se organizuje upotrebom konceptualnih modela kao što je UML. Poslovni proces se sastoji od aktivnosti čijim koordinisanim izvršavanjem se postiže željeni poslovni cilj. Aktivnosti se dijele na sistemske, interaktivne (korisničke) i manuelne. Određeni dijelovi poslovnog procesa se mogu izvršavati pod kontrolom "Workflow management" sistema. Pri modelovanju koriste se tri koncepta apstrakcije: vertikalna, horizontalna i agregaciona. Vertikalna apstrakcija ima nivo meta-metamodela, metamodela, modela i instance. Horizontalna apstrakcija obuhvata modelovanje funkcija, informacija, organizacije i IT okruženja. Modelovanje funkcija predstavlja modelovanje jedinica rada (units of work) koje se izvršavaju tokom poslovnog procesa, gruba podjela poslovnih funkcija, a fina podjela na operativnom nivou (aktivnosti). Modelovanje informacija obrađuje problem reprezentacije podataka u poslovnom procesu. Sve to zajedno čini model poslovnog procesa. Funkcije na visokom nivou, aktivnosti na niskom. Ponašanje svake instance aktivnosti se može opisati događajima – events. Svaki prelaz stanja instance aktivnosti se predstavlja kao jedan događaj. Stanje instance aktivnosti je opisano preko dva događaja -> događaj ulaska u trenutno stanje, i događaj izlaska iz trenutnog stanja. Model procesa predstavlja šablon za kreiranje instanci procesa, sastoji se od skupa modela aktivnosti. Model procesa se sastoji od čvorova koji mogu biti model aktivnosti, model događaja i model gateway-a, i usmjerenih grana koje predstavljaju veze između čvorova. Instance procesa imaju životni ciklus, sastoje se iz jedne ili više instanci aktivnosti. U modelovanju poslovnih procesa međuzavisnost podataka se reprezentuje tokom podataka između različitih aktivnosti.

Šabloni podataka u radnim procesima (Workflow data patterns) su uvedeni kako bi se organizovali podaci u upravljanju poslovnim procesima. Bazirani na konceptima

- Vidljivost podataka
- Interakcija podataka
- Prenos podataka
- Rutiranje procesa zasnovano na podacima

Vidljivost podataka se odnosi na:

- Task data (lokalni za određenu aktivnost)
- Block data (podaci vidljivi svim aktivnostima u okviru podprocesa)
- Workflow data (vidljivi svim aktivnostima u procesu)
- Environment data (podaci su dio okruženja u kojem se izvršava poslovni proces)

Interakcija podataka pokazuje kako se podaci mogu prosljeđivati između aktivnosti i procesa

- Između aktivnosti u istom procesu
- Između aktivnosti i podprocesa u istom procesu
- Između aktivnosti u različitim procesima
- Između poslovnog procesa i sistema za upravljanje poslovnim procesima

Prenos podataka objašnjava da se podaci mogu prenositi po vrijednosti ili po referenci.

Rutiranje na osnovu podataka -> vrijednost podataka može imati implikacije na izvršavanje procesa (koristi se za evaluaciju uslova u čvorovima odlučivanja).

Osnovni princip organizacionog modelovanja jeste resurs. Veza između organizacione strukture i poslovnog procesa ostvaruje se dodjelom radnih zadataka, koji predstavljaju instance aktivnosti dodijeljene određenim zaposlenim. Postoje:

- Direktna dodjela
- Dodjela po ulogama (izvršioci sami preuzimaju ponuđene zadatke ili se dodjeljuje jednom izvršiocu po nekom algoritmu)
- Odložena dodjela (odluka se donosi u toku izvršavanja procesa)
- Autorizacija (dodjela zadatka na osnovu izvršiočeve pozicije)
- Razdvajanje nadležnosti (omogućuje dodjelu zadaka različitim učesnicima)
- Obrada predmeta (case handling) -> zadaci se dodjeljuju uvijek istoj osobi
- Dodjela na osnovu prethodne istorije
- Organizaciona dodjela (koriste se stvarne pozicije korisnika)

Orkestracija procesa

Šabloni za kontrolu toka (Control flow patterns) predstavljaju osnovni alat za orkestraciju procesa. Nezavisni su od konkretnog jezika za opis procesa. Osnovni šabloni za kontrolu toka su (podržavaju ih svi jezivi za modelovanje procesa):

- Sekvenca
- "I" grananje

- "I" spajanje
- Ekskluzivno "ili" grananje
- Ekskluzivno "ili" spajanje

Ostali šabloni za kontrolu toka su:

- "ili" grananje
- "ili" spajanje
- Višestruko spajanje
- Diskriminator
- N od M spajanje
- Proizvoljni ciklusi
- Implicitna terminacija
- Višestruke instance (aktivnosti)
- Odloženo odlučivanje
- Sekvencijalno izvršavanje bez prethodnog znanja
- Milestone (referentna tačka)
- Šabloni za vrijeme izvršavanja procesa (runtime šabloni)

"ili" spajanje -> bez dodatnog znanja o samoj prirodi procesa nemoguće je reći šta je odgovarajuće ponašanje, npr ako se desi *terminate* događaj instance aktivnosti **b** (t_b), koja je reakcija čvora spajanja?

Višestruko spajanje (multi-merge) predstavlja tačku u procesu u kome se dve ili više konkurentnih niti procesa spajaju bez sinhronizacije. Aktivnost koja slijedi iza ovakvog tipa spajanja se instancira onoliko puta koliko je bilo aktiviranih dolaznih grana.

Diskriminator predstavlja tačku u procesu u kojoj se čeka da se izvrši jedna od ulaznih grana, prije nego što se aktivira naredna aktivnost. Nakon što je naredna aktivnost jednom aktivirana, diskriminator čeka da se sve ostale ulazne grane završe, ali ne instancira niti aktivira nove naredne aktivnosti. Nakon što su sve ulazne grane završile svoje aktivnosti, diskriminator se resetuje, tek nakon toga je moguće ponovno okidanje.

N od M spajanje predstavlja generalizaciju diskriminatora. Predstavlja tačku u procesu u kojoj se M paralelnih grana spaja u jednu. Čeka se da se izvrši $N \leq M$ ulaznih grana, prije nego što se aktivira naredna aktivnost. Kao i kod diskriminatora, kada se aktivnosti svih ulaznih grana terminiraju, čvor se resetuje.

Proizvoljni ciklusi predstavljaju mjesto u procesu u kome se jedna ili više aktivnosti može ponavljati. XOR grananje odlučuje da li se izvršava još jedan ciklus, ili se nastavlja sa izvršavanjem instance d. (petlja).

Implicitna terminacija iako definisana kao šablon za kontrolu toka, ona predstavlja uslov za završetak procesa. Instanca procesa bi trebala biti terminirana kada nema više ništa da se obavi.

Višestruke instance aktivnosti imaju tri vrste: bez sinhronizacije, sa prethodnim znanjem u momentu dizajna, i bez prethodnog znanja u momentu dizajna.

Višestruke instance aktivnosti bez sinhronizacije -> c krene odmah nakon što je zadnje b startano. Ovakav šablon narušava sekvencijalnost. Drugi problem je terminacija procesa.

Višestruke instance aktivnosti sa prethodnim znanjem u momentu dizajna modela predstavljaju šablon po kojem je broj potrebnih instanci određene aktivnosti poznat u momentu razvoja modela procesa. Instance aktivnosti su sinhronizovane, tako da se sledeća aktivnost omogućava tek kada su sve instance multiplicirane aktivnosti završene.

Višestruke instance aktivnosti bez prethodnog znanja u momentu dizajna modela predstavljaju šablon po kojem broj potrebnih instanci određene aktivnosti nije poznat u momentu razvoja modela procesa, niti u bilo kojoj fazi izvršavanja procesa pre nego se ova multiplicirana aktivnost omogući. Nove instance date aktivnosti mogu biti kreirane u momentu dok se postojeće još izvršavaju ili su neke već i završene, a sledeća aktivnost se omogućava kada se završi i poslednja instanca multiplicirane aktivnosti.

Odloženo odlučivanje predstavlja mesto u procesu gdje se bira jedna od mogućih grana izvršavanja. Izbor se ne vrši na osnovu neke vrijednosti podataka, ili na osnovu odluke korisnika, nego se okruženju ponudi nekoliko alternativa, i okruženje aktivira jednu putanju, a ostale grane se ne aktiviraju. Pošto se odlučivanje odlaže sve do momenta dok jedna grana nije aktivirana, ono je odloženo do posljednjeg momenta.

Sekvencijalno izvršavanje bez prethodnog znanja u vrijeme razvoja modela predstavlja šablon po kome se skup aktivnosti izvršava sekvencijalno po redoslijedu koji se utvrđuje u vrijeme izvršavanja procesa (za n aktivnosti n! mogućih kombinacija).

Milestone – referentna tačka je šablon koji se koristi kada se određena aktivnost omogući samo kada je dostignuta određena referentna tačka.

Runtime šabloni nisu dio modela procesa, karakterišu funkcionalnost okruženja za izvršavanje poslovnog procesa.

Orkestracija procesa – drugi dio

Petri mreže predstavljaju tehniku za specifikaciju poslovnih procesa, pri kojem se zanemaruje okruženje u kojem će se poslovni proces izvršavati. Osmislio ju je Carl Adam Petri, i predstavlja uopštavanje teorije automata uvođenjem konkurentnosti. Petri mreža se sastoji od:

- Mjesta (places) -> predstavljaju se krugovima
- Prelaza (transitions) - > predstavljaju se kvadratima
- Usmjerenih lukova – veza (directed arcs) -directed arcs) -> linije sa strelicama

Veze su isključivo moguće između mjesta i prelaza. Dinamika sistema se predstavlja konceptom žetona (tokena). Struktura Petri mreže je fiksna, tokeni mogu da se nalaze na različitim mjestima u mreži i time opisuju trenutno stanje mreže. Prelaz se okida kada je omogućen, odnosno kada u svim njegovim ulaznim mjestima postoji bar jedan token. Kada se desi prelaz, iz svakog ulaznog čvora se uklanja po jedan token, a u svaki izlazni se postavlja po jedan token. Ovo kretanje tokena se naziva igra tokena (token play). Obavljanje tranzicije mijenja stanje mreže, i predstavlja zapravo aktivnost, te tranzicije se smatraju aktivnim, dok su mjesta pasivna, i sa tokenima određuju trenutnu instancu

procesa. Pošto može postojati više instanci procesa, tokeni u Petri mreži ne moraju pripadati istoj, instanci procesa, ali je problem što Petri mreže nemaju ugrađen način za razlikovanje tokena. Petri mreže su uređena torka (P,T,F) gdje je P – konačan skup mjesta, T – konačan skup tranzicija, i F – relacija. Stanje Petri mreže definisano je funkcijom mapiranja $M:P \rightarrow N$ kojom se skup mjesta mapira na prirodne brojeve, uključujući 0 (1 ako mjesto ima token, 0 ako nema). Na osnovu ponašanja Petri mreža pri okidanju tranzicija, i strukturi tokena, razlikujemo sledeće Petri mreže:

- Mreže uslovnih događaja (osnovne Petri mreže)
- Mreže tranzicije mjesta
- Kolor Petri mreže

Kod mreža uslovnih događaja svako mjesto može sadržati tačno jedan token u jednom momentu, tokeni se ne mogu razlikovati.

Mreže tranzicije mjesta su proširenje osnovne klase, gdje se u svakom stanju Petri mreže u jednom mjestu može nalaziti proizvoljan broj tokena. Tranzicija je omogućena ako u ulaznom stanju postoji minimum onoliko tokena kolika je **težina** odgovarajuće grane. Tokeni se i dalje ne mogu odentifikovati.

Kolor Petri mreže uvode koncept boje, tokeni se mogu razlikovati. Omogućavanje tranzicije zavisi i od broja tokena i od njihove boje (vrijednosti).

Procesni lanci upravljani događajima se koriste za predstavu specifičnih domenskih koncepata, dosta neformalna notacija. Dio je Architecture of Integrated Information Systems (ARIS). ARIS house čine tri stuba – podaci, kontrole i funkcije. Krov predstavlja cijelu organizaciju. U svakoj zoni postoje tri nivoa apstrakcije:

- Konceptualni nivo (najviši nivo, model object value, procesni lanci, organizacioni dijagrami)
- Nivo arhitekture
- Implementacioni nivo

Modelovanje se zasniva na formiranju procesnih lanaca kojima upravljaju događaji koji predstavljaju pasivne elemente, na osnovu događaja se donose odluke, a funkcije su aktivni elementi. Osnovni simboli notacije su događaji, funkcije, konektori (neka vrsta gatewaya).

Workflow mreže predstavljaju unapređenje osnovnog koncepta Petri mreže. Za razliku od Petri mreža imaju ulaz i izlaz. Uvedeni su **triggeri** (okidači) kako bi se detaljnije opisale tranzicije.

Graf-bazirani jezici koji vode računa o zavisnosti podataka razvijeni u sklopu komercijalnog sistema, prikazuju međuzavisnost podataka između aktivnosti.

BPMN je razvijen od strane "Object management group", ima za cilj da objedini sve dobre prakse modelovanja u jednu specifikaciju, nalik na UML, i da premosti razlike između dizajna i implementacije, odnosno da tu notaciju razumiju kako menadžeri poslovnih procesa tako, i developeri koji implementiraju poslovni proces, do ljudi koji upravljaju i nadgledaju poslovne procese. Organizacioni aspekti se predstavljaju pomoću pool i swimlane koncepata. Elementi notacije su podijeljeni u četiri grupe:

- Flow objects – elementi kojima se gradi model
- Artifacts – dodatne informacije irelevantne za tok izvršavanja procesa
- Connection objects – povezuju elemente dijagrama
- Swimlanes – prikazuju organizacioni aspekt (uloge korisnika)

Aktivnosti predstavljaju jedinice "posla" koji treba obaviti u procesu, osnovni su element poslovnog procesa.

Događaji su spona između realnog svijeta i procesa koji će reagovati na događaj.

Event based gateway predstavlja šablon odloženog odlučivanja.

XML format je standardizovan što omogućava razmjenu modela između različitih aplikacija.

Koreografija procesa

Orkestracije procesa sarađuju razmjenom poruka. Koreografije obezbjeđuju interoperabilnost orkestracija procesa. Postoje inicijative da se koreografije standardizuju u određenim domenima, npr RosettaNet (nabavke), SWIFTNet (finansijske usluge), HL7 (zdravstvene usluge). Definišu se pravila koja učesnici moraju ispuniti kako bi učestvovali u međusobnoj saradnji. Ta pravila se definišu koreografijama. Faze razvoja su organizovane u dizajn koreografije i implementaciju koreografije. U svaku od faza, na različite načine, uključeni su:

- Inženjeri poslovnog procesa
- Sistem arhitekta
- Programeri (developeri)

Inženjeri poslovnog procesa su odgovorni za poslovne aspekte saradnje.

Sistem arhitekta su odgovorni za arhitekturu sistema. Učestvuju i u dizajnu i u implementaciji.

Programeri (developeri) su zaduženi za implementaciju modela koreografije.

//Isto bože sačuvaj

Manji dijelovi faze razvoja su:

- Identifikacija učesnika – služi da se definišu uloge učesnika u koreografiji
- Definicije kritičnih tačaka (milestones) – stanja koreografije u kojima su postignuti određeni rezultati, kritične tačke i njihov redoslijed definišu aspekte ponašanja koreografije (behavioural aspect)
- Identifikacija poruka – definišu se poruke koje se razmjenjuju, uzimaju se u obzir i poslovni (sadržaj) i tehnički (format) aspekti poruka
- Definicija koreografije – kombinuju se podaci iz prethodne dve faze
- Behavioural iterfaces – definišu interfejsa za svaku od uloga koja učestvuje u koreografiji

Scenario saradnje je specifikacija interakcije neophodna kako bi koreografija napredovala od jedne do druge tačke. Te interakcije koje se opisuju se dešavaju između dve kritične tačke. Taj model

interakcije se može predstaviti posebnim procesom, umjesto modela aktivnosti. Interfejsi ponašanja definišu pogled jednog učesnika na koreografiju.

Kompatibilnost predstavlja sposobnost skupa učesnika da uspješno sarađuju u skladu sa datom koreografijom procesa. Uzroci nekompatibilnosti su različiti formati poruka, i neusklađene interakcije. Postoje strukturna kompatibilnost i kompatibilnost ponašanja. Strukturna se odnosi na strukturu komunikacije gdje postoje slaba i jaka strukturna kompatibilnost. Kompatibilnost ponašanja uzima u obzir kontrolu toka između instanci procesa u konverzaciji, uzima se u obzir i orkestracija procesa učesnika u konverzaciji. "Martens" je prezentovao pristup za formalnu kontrolu kompatibilnosti ponašanja, a to su moduli radnih procesa (workflow modules) gdje se orkestracije procesa predstavljaju kao posebna klasa Petri mreža.

Workflow module -> svaka orkestracija procesa se predstavlja *workflow modulom* koji definiše njeno interno ponašanja, kao i spolja vidljivo komuniciranje.

Osobine poslovnih procesa

Postoji dva aspekta podataka: podaci nad kojima aktivnosti vrše neku manipulaciju (pozivanjem aplikacija ili servisa) u međuzavisnost podataka između različitih aktivnosti u procesu. Model procesa je strukturno ispravan ako postoji tačno jedan početni čvor, postoji tačno jedan krajnji čvor, a svi ostali čvorovi se nalaze na putanji od početnog ka krajnjem. Na ispravnost modela mogu da utiču deadlock-ovi (xor split, and join), livelock-ovi (beskonačne petlje). Analiza dostupnosti koristi grafove za provjeru ispravnosti, pogodna za manje modele. Za veće modele koriste se računarski sistemi za utvrđivanje ispravnosti. Za neke poluformalne metode modelovanja, kriterijum korektnosti je pretvrd, pa se uzima relaksirani kriterijum korektnosti. Relaksirani kriterijum korektnosti kaže da je model procesa prihvatljiv ako dozvoljava da se izvrše instance procesa koje dovode do željenog ishoda. Isto tako slabi kriterijum korektnosti se bazira na ideji da se sve funkcionalnosti koje neka orkestracija procesa nudi u okviru koreografije ne koriste.

Arhitektura sistema za upravljanje poslovnim procesima

Sistemi za upravljanje poslovnim procesima u sebi imaju sisteme za upravljanje poslovnim tokovima. Kod njih se jasno razlikuje vrijeme izrade poslovnog toka (build time) i vrijeme izvršavanja (run time). Model poslovnog toka mora se razviti tako da je u skladu sa operativnim procesom, proširuje operativni model poslovnog procesa detaljima neophodnim za izvršavanje. U zavisnosti od izabrane platforme predstavlja se kao izvršivi skript ili programski kod. Kada se izvršava poslovni proces, kreira se i instanca izvršivog poslovnog toka. Ne postoji veza između instance procesa i modela na osnovu kojeg je kreiran. Opšta arhitektura sistema za upravljanje poslovnim tokovima sadrži podsistem za modelovanje, repozitorijum modela i radno okruženje (workflow engine). U zavisnosti od toga da li se radi o automatskim poslovnim tokovima ili interaktivnim, nepohodan je ili podsistem za vezu sa drugim aplikacijama, ili GUI za interakciju sa korisnicima.