



Ivan Barać

Ivan Barać

Ericsson Nikola Tesla d.d., Zagreb, Hrvatska
Ericsson Nikola Tesla d.d., Zagreb, Croatia

Ključne riječi:

**Platforma za isporuku usluga, SDP
Okolina za kreiranje usluga, SCE
Uslužno orijentirana arhitektura, SOA
Web usluge
Davatelj usluga
Davatelj sadržaja
Parlay/OSA
Parlay X
Konvergencija**

Key words:

**Service Delivery Platform, SDP
Service Creation Environment, SCE
Service Oriented Architecture, SOA
Web Services
Service provider
Content provider
Parlay/OSA
Parlay X
Convergence**

Okolina za kreiranje usluga

Sažetak

Okolina za kreiranje usluga (SCE – *Service Creation Environment*) je skupina komponenti koje omogućavaju mrežnom operatoru isporuku novih digitalnih usluga putem horizontalno izgrađene uslužne mreže. Osnovna zadaća platforme za isporuku usluga (SDP – *Service Delivery Platform*) je podrška i integracija operatorovih poslovnih i operativnih procesa. Okolina za kreiranje usluga podržava cjeloviti životni ciklus usluge: kreiranje, izvedbu, aktivaciju, uporabu te naplatu i upravljanje uslugama i njihovo povlačenje iz uporabe.

Okolina za kreiranje usluga je dizajnirana za isporuku

usluga preko višestrukih pristupnih mreža. U tradicionalnim mrežama operatorova okolina za kreiranje novih usluga je operatoru omogućavala da sam kreira usluge ili da usluge osmisli u uskoj suradnji s nekoliko partnera, tzv. »walled garden» poslovni model. Međutim, operatori su počeli razmatrati i tražiti rješenja koja će im omogućiti partnerski odnos s puno širom grupom razvijatelja aplikacija/rješenja. Takav način rada zahtijeva da sučelja poslovna prema poslovnim (B2B - *Business to Business*), podrže uspostavu poslovnih odnosa između operatora i isporučitelja usluga te zajednički pristup mrežnim aplikacijskim uslugama koje je razvio partner.

Za brže i učinkovitije uvođenje novih usluga na tržište, B2B sučelja moraju partnerima omogućavati jednostavan rad, a operatorovo upravljanje procesima suradnje s velikim brojem partnera mora biti automatizirano.

Okolina za kreiranje usluga je mjesto susreta informatičkih i telekom svjetova. Neke od tehnologija koje su se pokazale pogodnima su, prije svega, aplikacijski poslužitelji za razvoj i implementaciju aplikacija te web usluge (*Web Services*) kao tehnologija za integraciju sučelja B2B i usluga unutar operatorove domene. Ericssonova platforma za isporuku usluga temelji se na navedenim tehnologijama i osigurava arhitekturu koja podržava operatora u njegovom rastu prema poslovnim modelima suradnje s velikim brojem partnera. To rješenje pruža zajedničku okolinu za razvoj i implementaciju korisničkih usluga, osiguravajući pristup mrežnim resursima za interne i vanjske razvijatelje aplikacija i davatelje usluga.

Service Creation Environment

Abstract

A Service Creation Environment (SCE) is a set of components that allows a network operator to deliver new digital services through a horizontal Service Network. A fundamental requirement is that the Service Delivery Platform (SDP) supports and tightly integrates with the operator's business and operational processes.

The Service Creation Environment supports complete service life-cycle management, including creation, deployment, provisioning, utilization and withdrawal, charging as well as operation and maintenance.

The Service Creation Environment is designed to deliver services over multiple access networks. Traditionally, the operators' environment for developing and launching new end-user services supported service creation by either in-house personnel or in close cooperation with several partners ("walled garden" business model). However, operators are now looking for solutions that will enable them to partner with a much larger group of application developers/providers. This puts new requirements on the Business to Business (B2B) interfaces used with regard to business cooperation of operators and their business partners, in particular with regard to providing access to the Service Creation Environment both to operators and their partners. To reduce the time, cost and complexity in bringing new end-user services to the market, the B2B inter-

faces must be easy-to-use for partners and the process of handling many partners must to a large degree be automated for the operator. The service creation environment is a meeting place for the IT and telecom worlds. Some important emerging technology winners here are application servers providing the development and deployment infrastructure for applications, and web services as the technology for B2B interfaces.

Ericsson's service delivery platform utilizes the above technologies and provides an architecture that supports the operator when growing into business models supporting mass partnering/mass application scenarios. The solution provides a common development and execution environment for end-user services, facilitating access to network capabilities for both internal and external application developers and service providers.

1. Uvod

Ericssonova okolina za kreiranje usluga (SCE – *Service Creation Environment*) i platforma za isporuku usluga (SDP – *Service Delivery Platform*) predstavljaju jedinstveno rješenje za uslužne mreže, omogućavajući potpunu konvergenciju usluga bez obzira na način pristupa i vrstu korisnikovoga terminala. U promjenjivom svijetu komunikacija, prateći korisničke potrebe i u skladu s Ericssonovom vizijom sveprisutne komunikacije, platforma za isporuku usluga svojom otvorenosću, arhitekturom, dizajnom i mogućnostima u potpunosti ispunjava očekivanja i potrebe mrežnih operatora, davatelja usluga, sadržaja i krajnjih korisnika tih usluga. Ostvarivanje tako kompleksne zadaće nije moguće bez cjelovitoga pristupa komunikacijskim rješenjima koji je karakterističan za Ericssonove tehnologije i tradiciju.

Potreba omogućavanja pristupa mrežnim mogućnostima za razvijatelje aplikacija i davatelje usluga postaje ključna kao osnova za povećavanje mrežnog prometa i prihoda operatora. Platforma za isporuku usluga koja ispunjava zahtjeve za brzim razvojem i uvođenjem novih usluga ključni je faktor uspjeha u toj nakani.

Ovaj članak ima namjeru upoznati čitatelje s kompleksnošću i različitim potrebama takve platforme, ali i ostalih protagonista u današnjem poslovnom okruženju, ne samo telekom i IT industrije, već i drugih djelatnosti, a prije svega medija i oglašavanja. U drugom poglavlju dat je opis poslovnih procesa i mogućih scenarija interakcija između zainteresiranih strana (operator, razvijatelj aplikacije, isporučitelj sadržaja, davatelj usluga...).

Treće poglavlje daje pregled funkcionalnih zahtjeva i logičke arhitekture okoline za razvoj i isporuku usluga.

U četvrtom poglavlju opisani su principi, obrasci te arhitekturna rješenja i tehnologije korišteni u okolini za kreiranje usluga.

2. Poslovni modeli i uloge pojedinih sudionika u kreiranju usluga

2.1. Poslovne uloge

Protagonisti uključeni u kreiranje usluga za krajnjeg korisnika mogu se pojavljivati u različitim ulogama. Te uloge su osnova opisa suvremenih poslovnih modela u poglavlju 2.2 ovoga članka.

Slika 1. prikazuje glavne poslovne uloge u okolini za kreiranje usluga i odnose među njima.

2.2. Opis poslovnih uloga u okolini za kreiranje usluga

2.2.1. Krajnji korisnik

Krajnji korisnik (EU – *End User*) je potrošač usluga, aplikacija i mrežnoga pristupa. Krajnji korisnik može biti organizacija ili pojedinac (individualni korisnik). Poseban slučaj krajnjeg korisnika je aplikacija, npr. aplikacija koja koristi mrežnu uslugu pozicioniranja za lociranje krajnjeg korisnika.

Individualnim krajnjim korisnicima važan je odnos povjerenja sa sakupljačem sadržaja i aplikacija (CAA - *Content and Application Aggregator*) te zaštita njegove privatnosti i zaštita od neželjenog sadržaja (*spam*). Za osiguranje takvih uvjeta za krajnjega korisnika važna je podrška za aktivno prihvaćanje usluge na koju je korisnik pretplaćen.

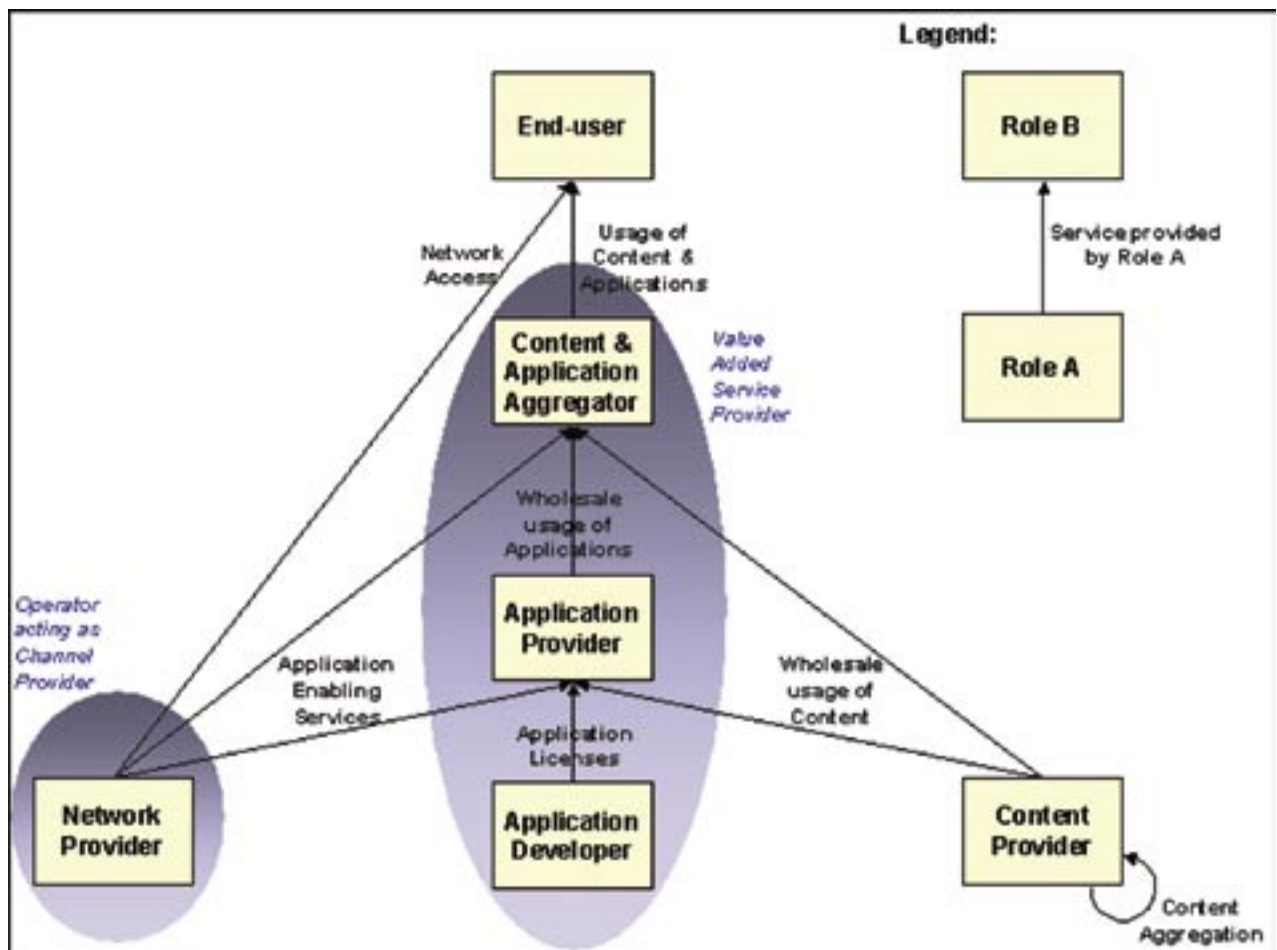
2.2.2. Mrežni operator

Važno je naglasiti da je mrežni operator (NP - *Network Provider*) u realnosti skup nekoliko poslovnih uloga: davatelj pristupa, davatelj platnih usluga, davatelj identiteta (*access provider, payment provider, identity provider*), i slično. Međutim, s obzirom na to da se na tržištu operator skoro uvijek javlja u svim tim ulogama, zbog jednostavnosti opisan je ovdje kao protagonist u jednoj ulozi: mrežni operator.

Mrežni operator osigurava:

1. pristup nizu komunikacijskih usluga krajnjim korisnicima;
2. različite mrežne aplikacijske usluge, kao što su lokacija, prisutnost, kontrola poziva i slično, isporučiteljima aplikacija (*Application Providers*);
3. specifične komponente i usluge sakupljačima sadržaja i aplikacija krajnjim korisnicima. Primjeri takvih usluga su identitet korisnika i tarifiranje.

Možemo reći da su dvije najvažnije vrijednosti koje vezujemo uz mrežnoga operatora komunikacijska infrastruktura te kvalitetan odnos sa svim krajnjim korisnicima mreže.



Slika 1. Glavne poslovne uloge u okolini za kreiranje usluga

2.2.3. Sakupljač sadržaja i aplikacija

Sakupljač sadržaja i aplikacija osigurava krajnjem korisniku pristup do brojnih usluga s dodanom vrijednošću (VAS - *Value Added Service*), kao što su sadržaji (npr., *Streaming*, spremanje sadržaja), interaktivne usluge (igranje, prisutnost), osobne usluge (bankarstvo), komunikacija (*e-mail*), m-trgovina (*m-commerce*) i slično.

U većini slučajeva ovo nije jedina uloga tog protagonista. Moguće je kombinirati tu ulogu s ulogom mrežnoga operatora ili s onom isporučitelja aplikacija.

2.2.4. Isporučitelj aplikacija

Isporučitelj aplikacija (AP - *Application Provider*) se brine za upravljanje ponudom aplikacija prema krajnjem korisniku te prodaje korištenje aplikacija na komercijalnim principima.

2.2.5. Razvijatelj aplikacija

Razvijatelj aplikacija (AD - *Application Developer*) je protagonist koji razvija i testira aplikacije te prodaje licence za

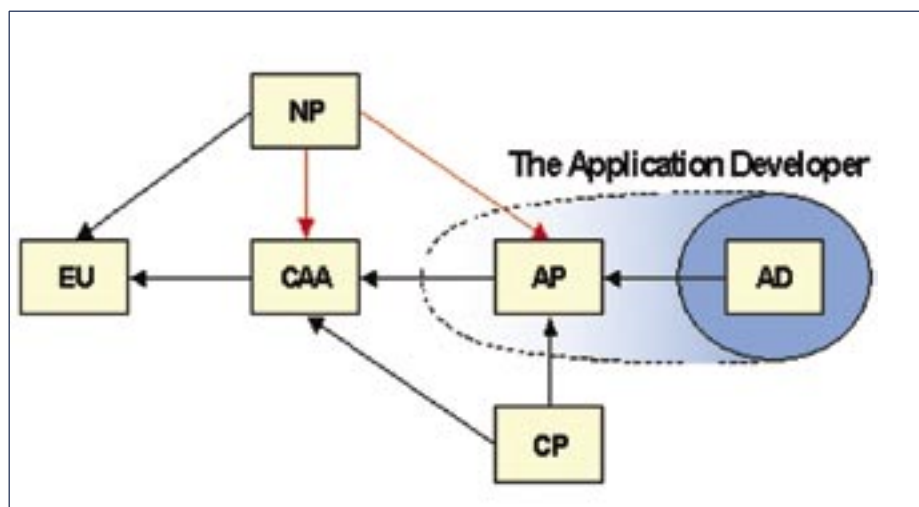
korištenje aplikacija. Razvijatelj aplikacija je najčešće mali ili srednji neovisni davatelj aplikacijske podrške (ISV - *Independent Software Vendor*), ali može biti bilo tko, uključujući i velike kompanije ili samo jednog čovjeka.

Važno za napomenuti je činjenica da u procesu razvoja aplikacije sam razvijatelj aplikacije mora privremeno preuzeti ulogu isporučitelja aplikacija, kao što je prikazano na Slici 2. Zaključak je da je mrežnom operatoru potrebna suradnja i s razvijateljima aplikacija radi osiguranja jednostavnijega uključivanja aplikacijskih mrežnih usluga u proces razvoja aplikacija za krajnjega korisnika.

Da bi operator mogao surađivati s velikim brojem razvijatelja aplikacija postoji nekoliko preduvjeta o kojima se mora voditi računa.

Na primjer, velika većina razvijatelja aplikacija:

- ima malo iskustva u specijaliziranim tehnologijama korištenim za razvoj mobilnih aplikacija;
- ima ograničene mogućnosti i interes za dugotrajne pregovore za uspostavu odnosa s operatorima;
- traži pomoć pri verifikaciji aplikacija u operatorovom okruženju;



Slika 2. Pozicija razvijatelja aplikacija

2.3. Poslovni modeli i protagonisti

Na tržištu različiti protagonisti nastupaju u sklopu jedne ili više uloga, opisanih u prethodnom poglavlju, u zavisnosti od trenutnog poslovnog modela (*Business Models and Actors*). Važno je naglasiti da različiti poslovni modeli nisu međusobno isključujući. Naprotiv, u većini slučajeva nekoliko poslovnih modela se koristi istovremeno, ovisno o potrebama same usluge koja je ponuđena krajnjem korisniku.

Neki od poslovnih modela baziranih na ulogama definiranim u poglavlju 2.2 su opisani u nastavku. Uloge su opisane s gledišta kupca platforme za isporuku usluga, tj. mrežnoga operatora.

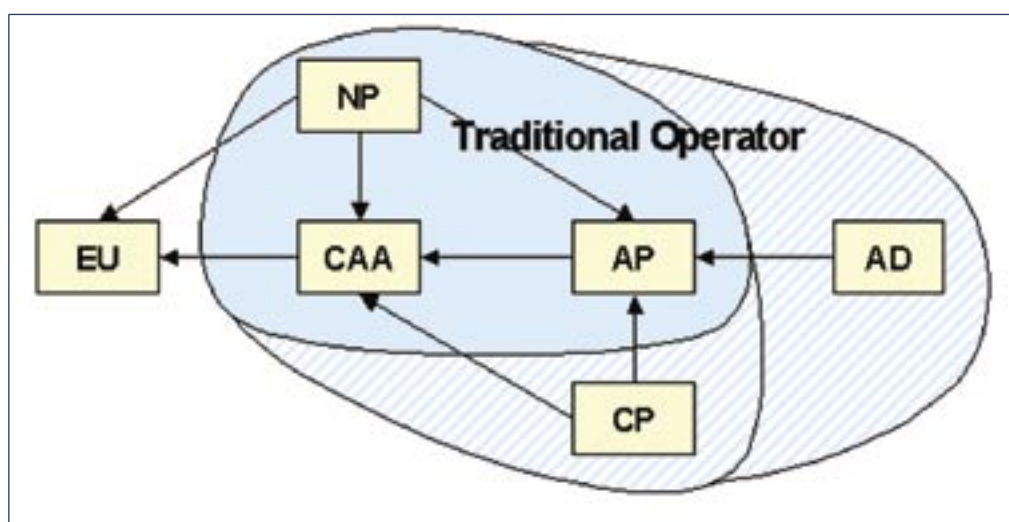
Mrežni operator osigurava mrežnu infrastrukturu, međutim, može preuzeti i neku od drugih uloga.

2.3.1. Operator kao tradicionalni davatelj usluga

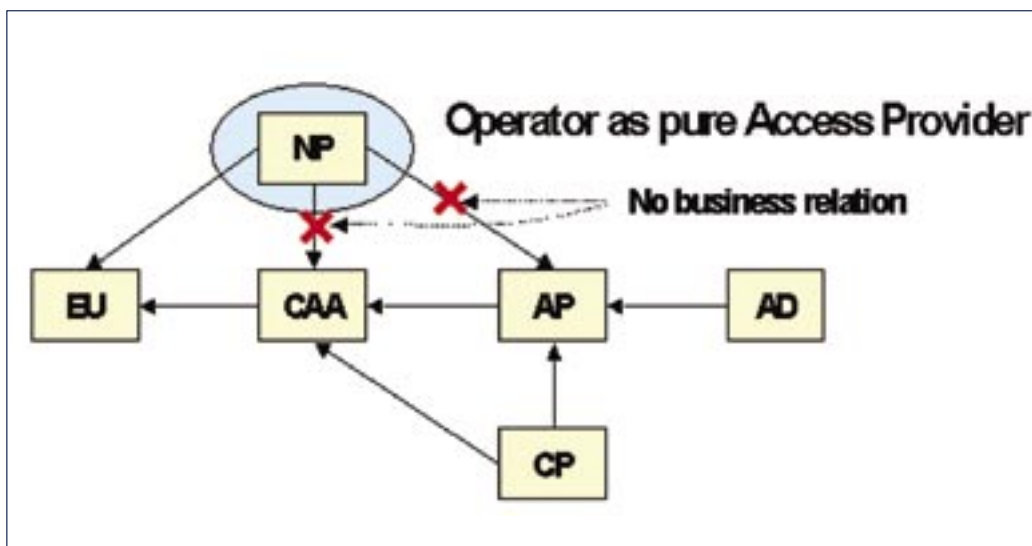
Povijesno gledano, telekom operatori su bili jedini davatelji usluga krajnjim korisnicima, povijesni model (*Operator as Traditional Service Provider: The Historical Model*), odnosno, osim osiguranja mrežne infrastrukture, operator je preuzimao ulogu sakupljača sadržaja i aplikacija, isporučitelja aplikacija, a često i sadržaja, te razvijatelja aplikacija (Sl. 3).

Prednosti operatora su sljedeće:

- interni (neformalni) odnosi u pružanju mrežnih i usluga i usluga podrške, koji olakšavaju osiguravanje sigurnosti, plaćanje i slično;
- potpuna zarada od sadržaja i aplikacija ostaje operatoru.



Slika 3. Tradicionalni davatelj usluga



Slika 4. Operator kao "čisti" davatelj pristupa

Međutim, nemogućnost da se ostane davatelj svih usluga koje žele krajnji korisnici u postojećem modelu je jasan komparativni nedostatak prema konkurenciji.

2.3.2. Operator kao davatelj pristupa

Jedan od uvriježenih poslovnih modela danas je svakako onaj u kojemu je operator davatelj pristupa krajnjim korisnicima bez pružanja mrežnih usluga ili usluga podrške vanjskim partnerima, Internet model usluga (*Operator as Pure Access Provider: The Internet Services Model*). Tipičan primjer takvoga modela su usluge koje korisnici pronalaze na Internetu.

U ovom slučaju (Sl. 4.) prednost za operatora predstavlja

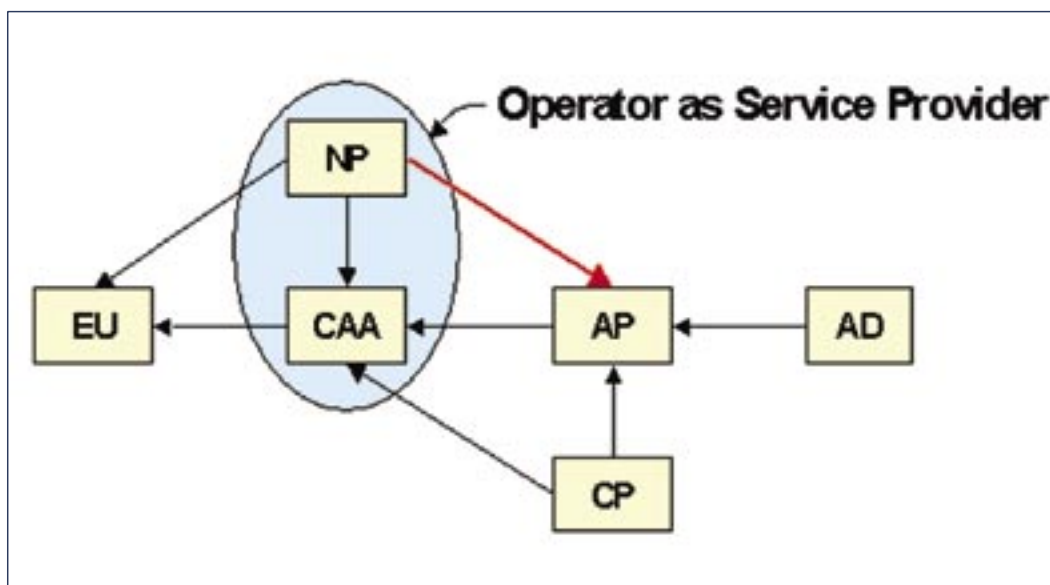
činjenica da krajnji korisnik može pristupiti sadržajima i aplikacijama na Internetu, tako povećavajući promet, a samim tim i prihode davatelja pristupa.

Nedostaci su sljedeći:

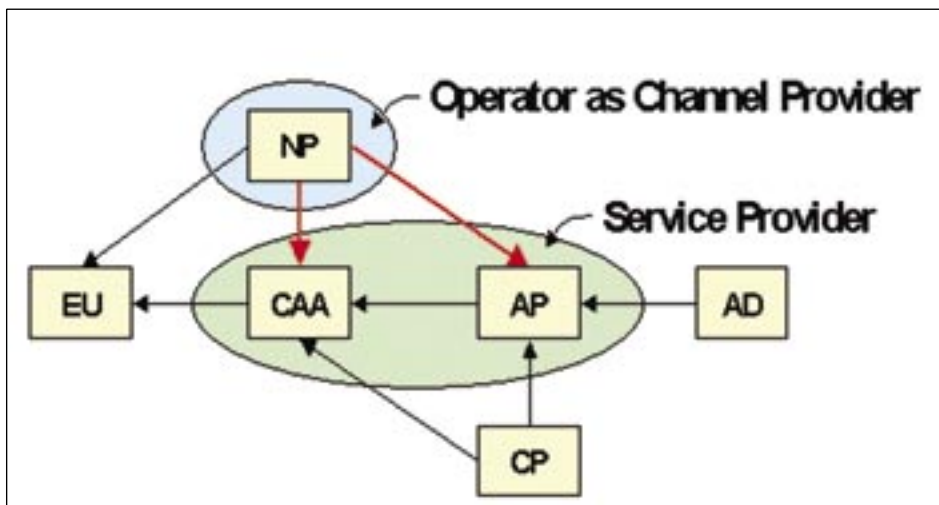
- operator nije dio u vrijednosnom lancu aplikacija i sadržaja (nema podjele dobiti);
- aplikacije su limitiranih funkcionalnosti zbog nemogućnosti korištenja mrežnih usluga;
- potencijali operatora da isporuči mrežne usluge i/ili usluge podrške nisu iskorišteni.

2.3.3. Operator kao davatelj usluga

Neki značajni svjetski operatori danas podržavaju poslovni



Slika 5. Operator kao davatelj usluga



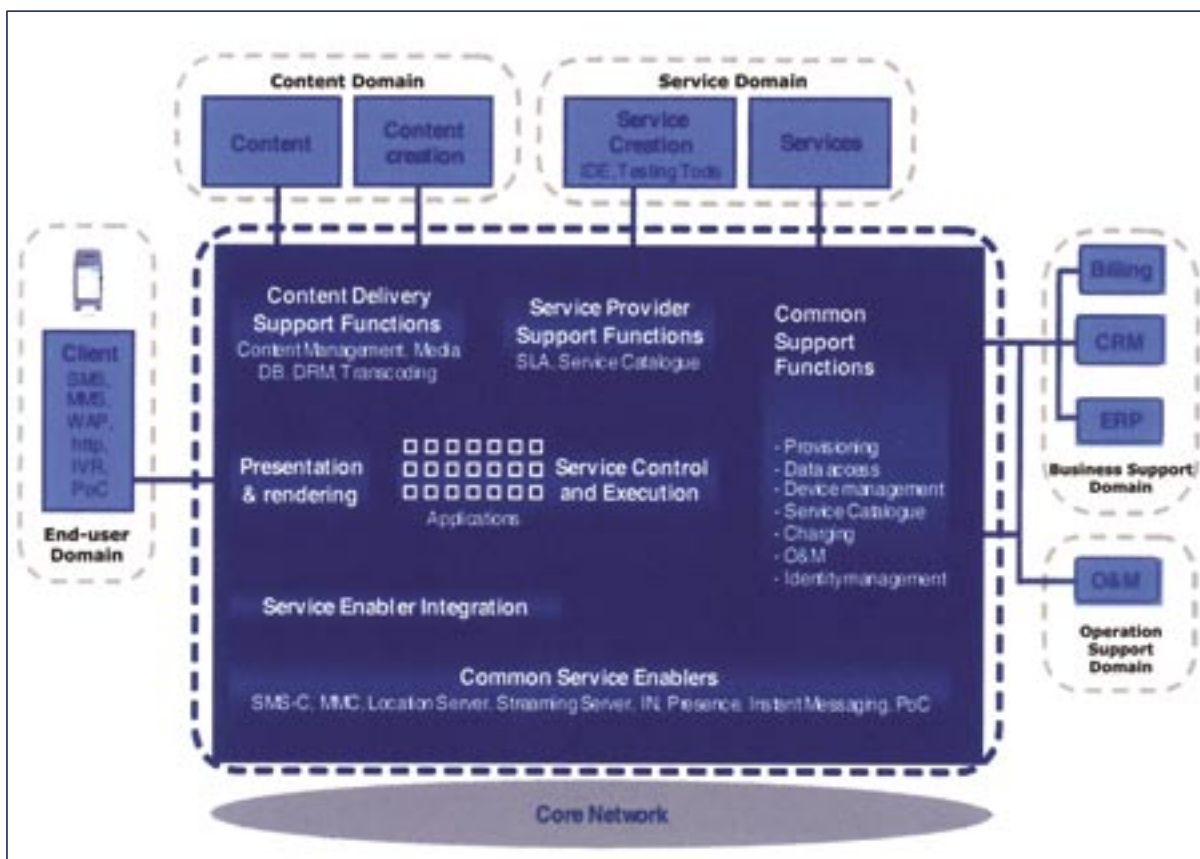
Slika 6. Operator kao davatelj kanala

model u kojem oni promoviraju usluge za krajnje korisnike, a poslovni partneri razvijaju i održavaju aplikacije koje omogućavaju ponuđene usluge (*Operator as Service Provider: The Branded Services Model*).

Operatori danas koriste i tzv. “Walled garden” model pri kojem, uz ugovor o razini usluge (SLA - *Service Level Agreement*), s malim brojem isporučitelja aplikacija ugovaraju zajednički marketing usluga.

Međutim, tehničke i administrativne barijere za uspostavu partnerskih odnosa su kompleksne i često su formalizirane u obliku procesa kvalifikacija. Drugim riječima, usluge ponuđene preko sučelja prikazanog crvenom strelicom na Slici 5. su nedostupne velikoj većini potencijalnih poslovnih partnera danas.

Prednosti za operatora kao davatelja usluga su sljedeće:



Slika 7. Platforma za isporuku usluga - logički prikaz

- operator kontrolira vrijednosni lanac sadržaja i aplikacija;
- operator ima mogućnost pružanja mrežnih usluga vanjskim partnerima za uporabu u njihovim aplikacijama.

Nedostaci ovoga modela su:

- mrežne usluge pružene partnerima su limitirane i nezgrapne te ih partneri ne mogu dovoljno učinkovito koristiti;
- poslovna sučelja prema isporučiteljima aplikacija su izgrađena na paradigmi “mali broj - veliki, dobro poznati isporučitelji”, što isključuje veliki broj isporučitelja aplikacija i samim tim smanjuje broj usluga koje se mogu nuditi krajnjim korisnicima.

2.3.4. Operator kao davatelj kanala

Operatori su se zadnjih godina sve učestalije počeli pojavljivati kao davatelji kanala, model potpune usluge (*Operator as Channel Provider: The Full Services Model*). To znači da operator pruža velikom broju davatelja usluga ne samo mogućnost korištenja mrežnih usluga, kao što su kontrola poziva, prisutnost, lokacija krajnjeg korisnika, i slično, već im također pruža i usluge podrške vezane uz upravljanje iden-

titetom (*Identity Management*): autentifikacija, autorizacija, upravljanje sustava tarifiranja, i slično. Poslovna sučelja označena crvenom strelicom na *Slici 6*. tako postaju sve važnija.

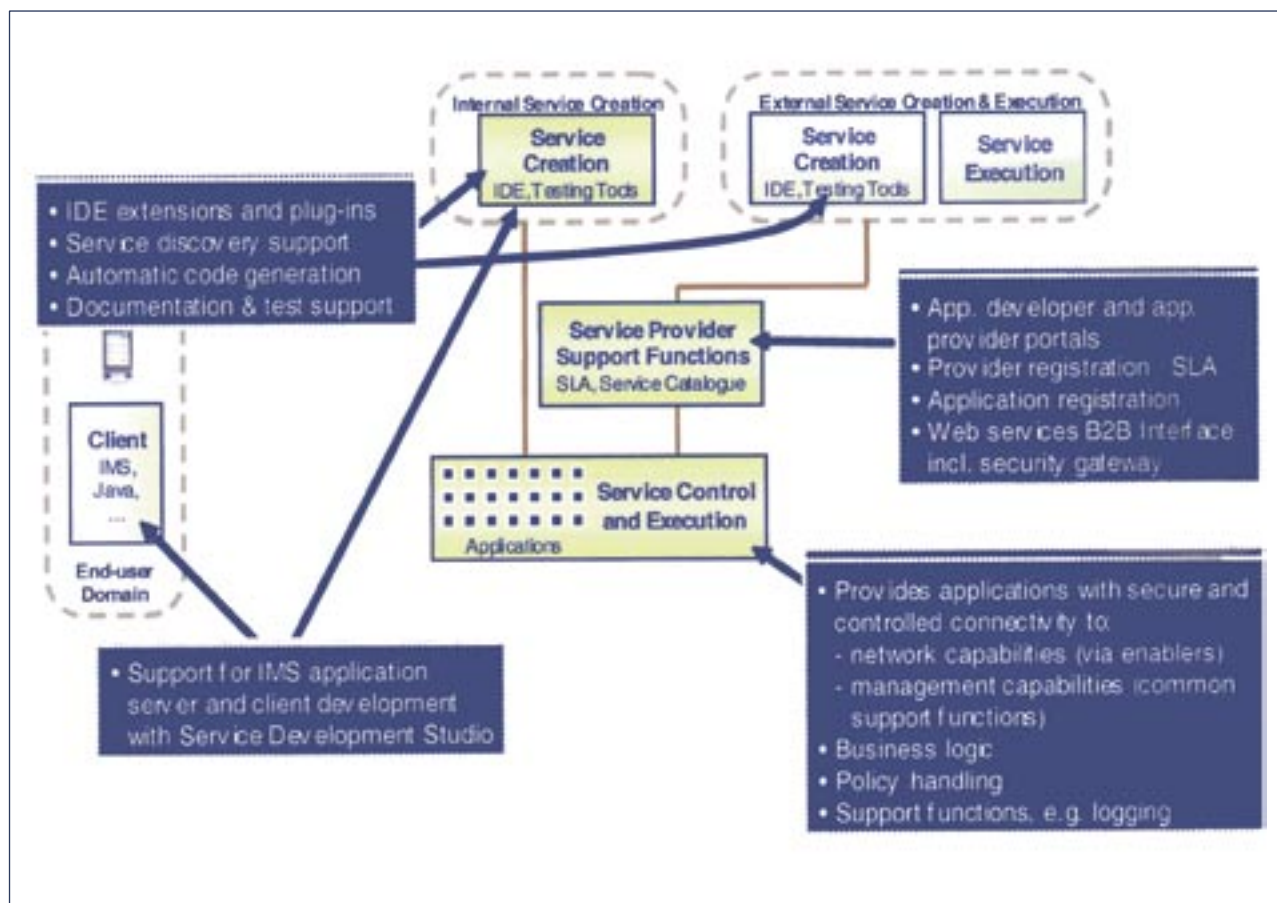
Prednosti za operatore su sljedeće:

- operator posjeduje veći dio vrijednosnog lanca sadržaja i aplikacija, što mu osigurava povećanje prihoda;
- operator može svojim poslovnim partnerima pružiti mrežne usluge za korištenje u njihovim aplikacijama (“a network in every application”);
- operator može svojim poslovnim partnerima osigurati usluge podrške.

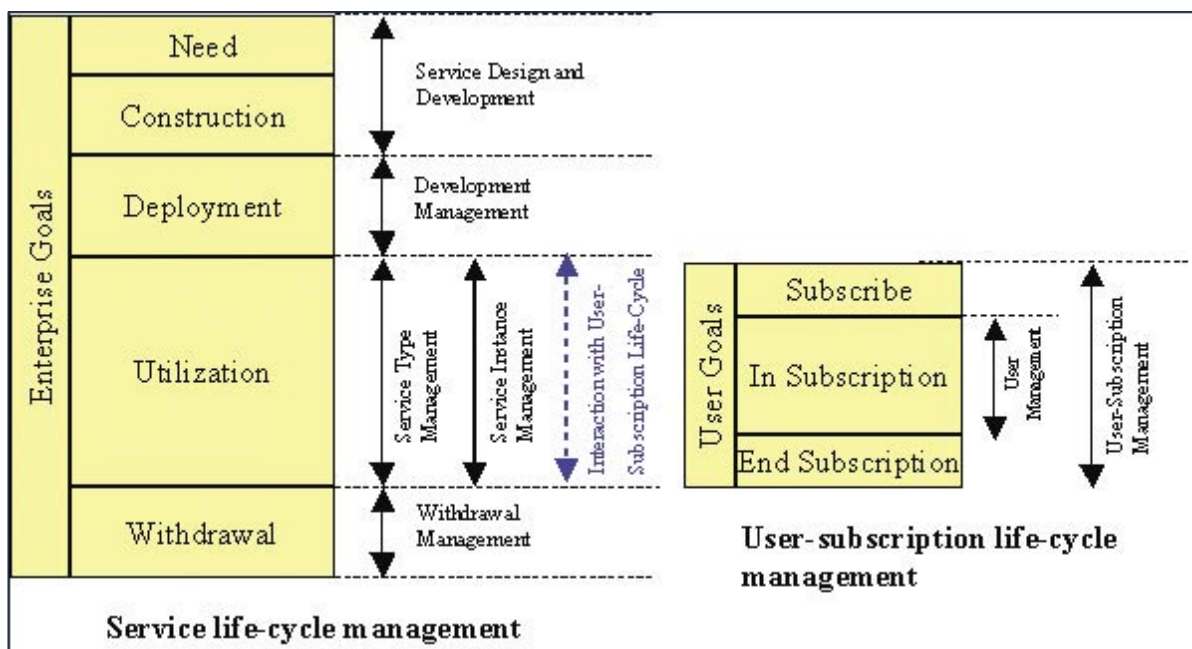
Nedostatak ovoga modela sastoji se u ograničenim prihodima od sadržaja i aplikacija u usporedbi s modelom u kojemu ih operator nudi sam.

2.4. Fleksibilna podrška poslovnih modela

Kao što je već spomenuto, operatori ne koriste samo jedan od mogućih modela, nego posežu za onim koji je najpogodniji za svaku specifičnu uslugu i specifičnu grupu krajnjih korisnika. Jedna od najvećih prepreka postizanja željene fle-



Slika 8. Prikaz okoline za kreiranje i isporuku usluga



Slika 9. Upravljanje životnim ciklusom usluge i korisničkom pretplatom na uslugu po TINA-C modelu

ksibilnosti je potreba za kompleksnom i zahtjevnom integracijom specifičnom za telekom industriju, pa se cilj koji se postavlja može realizirati samo uz pomoć platforme za isporuku usluga koja će moći podržati i buduće potrebe (*future-proof platform*).

2.4.1. Prednosti za operatora

Takva platforma operatorima osigurava brojne ključne funkcionalne prednosti koje osnažuju njegovo poslovanje. U nastavku su navedene prednosti ne samo za mrežne operatore, već i za sakupljače aplikacija i sadržaja te isporučitelje aplikacija, s obzirom na to da će operator u manjoj ili većoj mjeri također djelovati u svakoj od tih uloga:

- skraćeno vrijeme plasiranja usluge na tržište (*shorter TTM, shorter Time to Market*) od koncepta usluge do plasiranja kroz korištenje komercijalnih aplikacijskih poslužitelja, razvojnih alata i pristup širokoj zajednici razvijatelja aplikacija;
- optimalna naplata usluga krajnjim korisnicima korištenjem fleksibilnih modela naplate, npr. naplata po događaju, pretplati, sadržaju, i slično;
- status svake usluge ponuđene krajnjem korisniku je poznat u svakom trenutku (*service assurance*);
- veliki broj davatelja usluga s kojima se uspostavljaju odnosi putem automatiziranih procesa i sučelja, npr. ugovora o razini usluge;
- okruženje u kojemu je povezano više različitih isporučitelja (*multivendor environment*) upravljano arhitekturom koja podržava jednostavnu adaptaciju različitih sučelja.

2.4.2. Prednosti za razvijatelje aplikacija

- skraćeno vrijeme za uključivanje mrežnih usluga, što je omogućeno jednostavnim sučeljima i pristupom te standardnim razvojnim alatima;
- jednostavan pristup mrežnim uslugama operatora, bez potrebe za kompleksnim ugovorima.

2.4.3. Prednosti za krajnjeg korisnika

- jednostavan pristup, korištenje i upravljanje velikim brojem usluga, bez obzira na vrstu i tip sadržaja i aplikacija;
- rast broja i spektra usluga zahvaljujući troškovno učinkovitom razvoju, izvedbi i izvršavanju novih aplikacija;
- povoljnije cijene uslijed tržišne konkurencije davatelja usluga.

3. Koncept platforme za isporuku usluga

Konceptualna logička arhitektura platforme za isporuku usluga (SDP - *Service Delivery Platform*) prikazana je na Slici 7. Može se uočiti da je uslužna mreža podijeljena u niz logičkih funkcija koje čine jedinstvenu cjelinu sposobnu ispuniti poslovne zahtjeve operatora u ulozi davatelja kanala tijekom životnoga ciklusa usluge.

Platforma za isporuku usluga je cjelovito rješenje koje podržava kompletni životni ciklus usluge i može se podijeliti u tri konceptualne cjeline koje u sebi sadržavaju potrebne funkcije:

- **Okolina za kreiranje i isporuku usluga**
- funkcije podrške sadržajima
- funkcije podrške davateljima usluga
- prezentacija i prikaz sadržaja
- aplikacije
- **Zajedničke funkcije podrške**
- aktivacija usluga
- pristup podatcima
- katalog usluga
- naplata usluga
- održavanje i upravljanje uslugama
- upravljanje identitetom korisnika i partnera
- **Zajednički integracijski sloj**
- OSA/Parlay pristupnik
- SMS centar
- MMS centar
- poslužitelj za instant poručivanje
- streaming poslužitelj
- poslužitelj za mobilno pozicioniranje
- poslužitelj prisutnosti.

Detalniji prikaz okoline za kreiranje i isporuku usluga nalazi se na *Slici 8.* te se naglasak stavlja na tehnološka i poslovna rješenja toga dijela cjelokupne platforme za isporuku usluga. U nastavku članka je opisana okolina za kreiranje i isporuku usluga, a navedeni su i utjecaji različitih aspekata na nju.

4. Kreiranje i isporuka usluga

4.1. Koncept životnog ciklusa usluge

Kod kreiranja usluge valja imati na umu slijedeće faze životnog ciklusa usluge:

- Razvoj (*development*)
- Izvedba (*deployment*)
- Uporaba (*utilization*)
- Povlačenje iz uporabe (*withdrawal*).

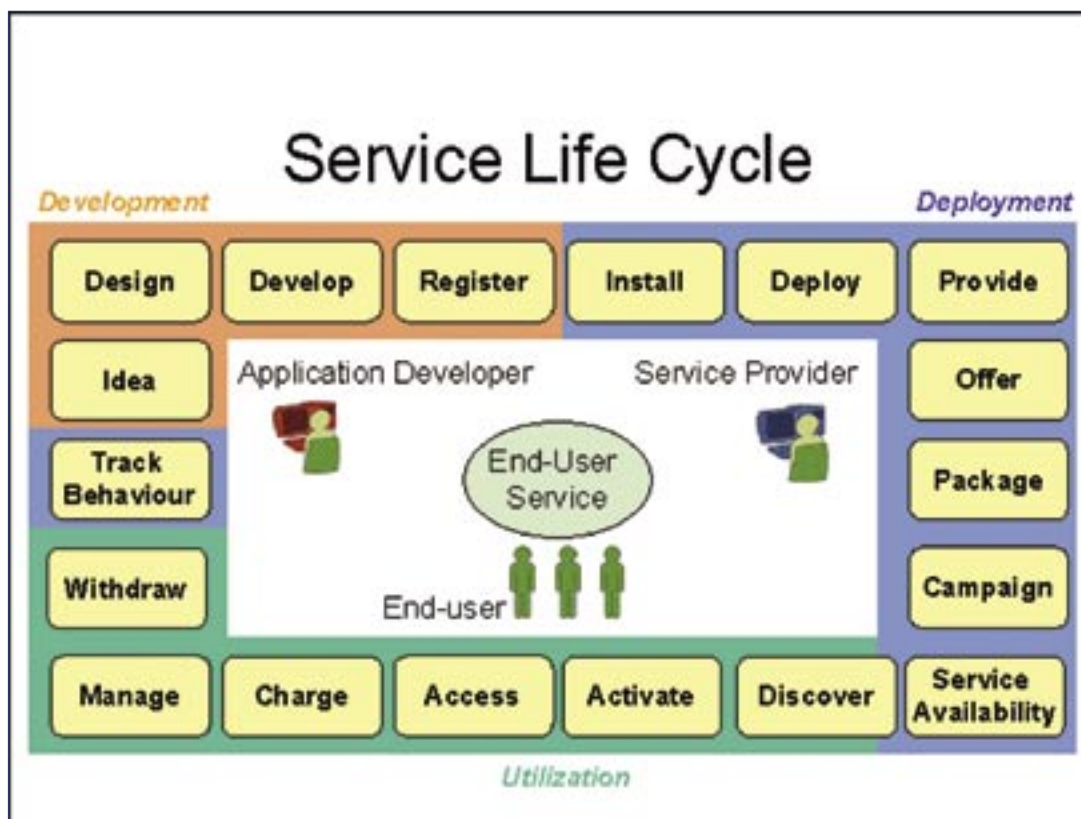
Ovi termini su usklađeni s TINA-C (*Telecommunications Information Networking Architecture Consortium*) modelom usluga. Životni ciklus usluge i korisničke pretplate na uslugu je prikazan na *Slici 9.*

TINA-C model, prikazan na *Slici 9.*, je samo grubi model koji se može detaljnije razraditi, što je prikazano na *Slici 10.*

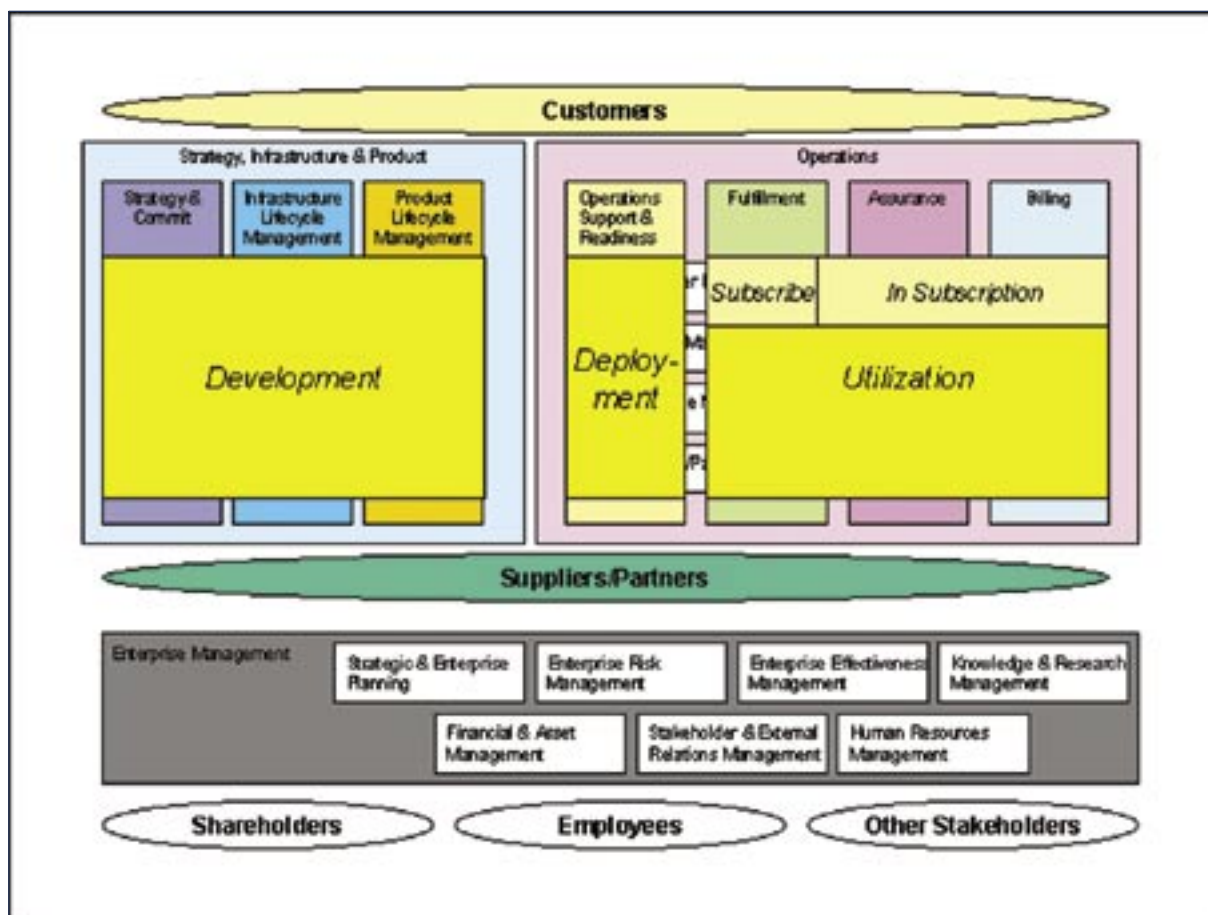
Termini korišteni u detaljnoj razradi životnog ciklusa usluge mogu se mapirati u eTOM procese kako slijedi (*Sl. 11.*):

• UPRAVLJANJE ŽIVOTNIM CIKLUSOM USLUGE (SERVICE LIFE CYCLE MANAGEMENT):

- razvoj (*development*) je dio eTOM (*enhanced Telecom Operations Map*) upravljanja životnim ciklusom usluge (*Product Life Cycle Management*)
- izvedba i povlačenje iz uporabe (*Deployment & Withdraw-*



Slika 10. Detaljnija razrada životnog ciklusa usluge



Slika 11. Upravljanje životnim ciklusom usluge i korisničke pretplate na uslugu i eTOM mapa

al) je dio eTOM procesa *Operations Support & Readiness*

- uporaba (*utilization*) je dio eTOM procesa *Fulfillment, Assurance and Billing*

- **UPRAVLJANJE ŽIVOTNIM CIKLUSOM KORISNIČKE PRETPLATE NA USLUGU (USER-SUBSCRIPTION LIFE CYCLE MANAGEMENT)**

- pretplata (*subscribe*) je dio eTOM procesa *Fulfillment*

- pretplaćen (*in subscription*) je dio eTOM procesa *Assurance and Billing*

4.2. Okolina za kreiranje usluga i tehnološke domene

Konvergencija telekoma i datakoma se razmatra već duži niz godina. Slika 12. prikazuje rezultat jednoga od takvih pokušaja (Yankee Group).

U ovom konceptu konvergencija je opisana kao sukob. Do određenih razmjera to je i istina. Usprkos tome različiti telekom i datakom proizvođači vide izlaz iz ovoga sukoba u rješenjima koja su, uz objedinjavanje prednosti iz oba područja, puno efikasnija nego prijašnja specifična rješenja iz pojedine domene.

Ericsonovo rješenje je prikazano na Slici 13. Takvo rješenje u potpunosti iskorištava tehnologije i infrastrukturu koju mogu osigurati datakom kompanije za otvaranje telekom

specifičnih mogućnosti prema informacijskim i komunikacijskim aplikacijama koje ih koriste za pružanje usluga krajnjem korisniku. Okolina za kreiranje usluga je softversko rješenje koje primjenjuje moderne koncepte, principe, obrasce i najbolja iskustva. Sljedeća poglavlja će objasniti i dati naputke i primjere takvog generalnog i krosindustrijskog pristupa i tehnologija.

4.2.1. Distribuirana višeredna arhitektura aplikacijske podrške

Višeredni pristup je način strukturiranja aplikacija (aplikacijskog sloja). Takav se način, na primjer, koristi za tzv. *front-end/presentation/facade* specifične dijelove, *generic/core/reusable* dijelove poslovne logike i *back-end/data/resource connection* specifične dijelove kao što je prikazano na Slici 14. Distribuirana višeredna arhitektura programske podrške je vrlo poznati princip i najbolja praksa te se primjenjuje do određene razine u svakoj aplikaciji programske podrške. Ovaj koncept temelji se na razdvajanju interesa, (*separation of concern*) obrascu koji podržava lakše održavanje i proširavanje programske podrške, u isto vrijeme promičući ideju ponovne uporabe pojedinih (*re-use*) dijelova programske podrške.

Redni pristup može se primijeniti i na logički i na fizički dio

rješenja. Također je važno uočiti da cjelokupna struktura redova može biti posložena u kaskadu, kao, na primjer, kada je određeni aplikacijski sustav dio drugog većeg aplikacijskog sustava.

Komponente u svakom redu, a posebice u redu temeljne logike (*core logic*) mogu se također slagati u kaskadu interno u istom sloju. Generalna struktura svakog reda može se primijeniti rekurzivno na različite razine sustava/komponenti, od najvišeg logičkog dijela ka nižim logičkim dijelovima sustava, pa do najmanjih komponenti, ukoliko strukturiramo rješenje po principu prezentacijskog reda fasade, temeljnog dijela poslovne logike i dijela spoja prema resursima.

Iako nazivi samih redova mogu ovisiti o rječnicima korištenim u različitim domenama ili specifičnim implementacijama u pojedinim tehnološkim domenama, pristup koji podržava principe i posjeduje komponente temeljne logike, *front-end* adaptacije/medijacije prema klijentima i *back-end* adaptacije/medijacije prema resursima je sličan za sve tipove sustava za aplikacijsku podršku (*software systems*).

Zahtjevi koji se mogu postaviti za svaki red su:

- prezentacijska fasada (*Presentation Façade*): komponente izlažu usluge putem *web* portala (prilagođeno različitim korisničkim segmentima), komponente izlažu usluge kao

web usluge (prilagođene različitim standardima); sigurnost, itd.;

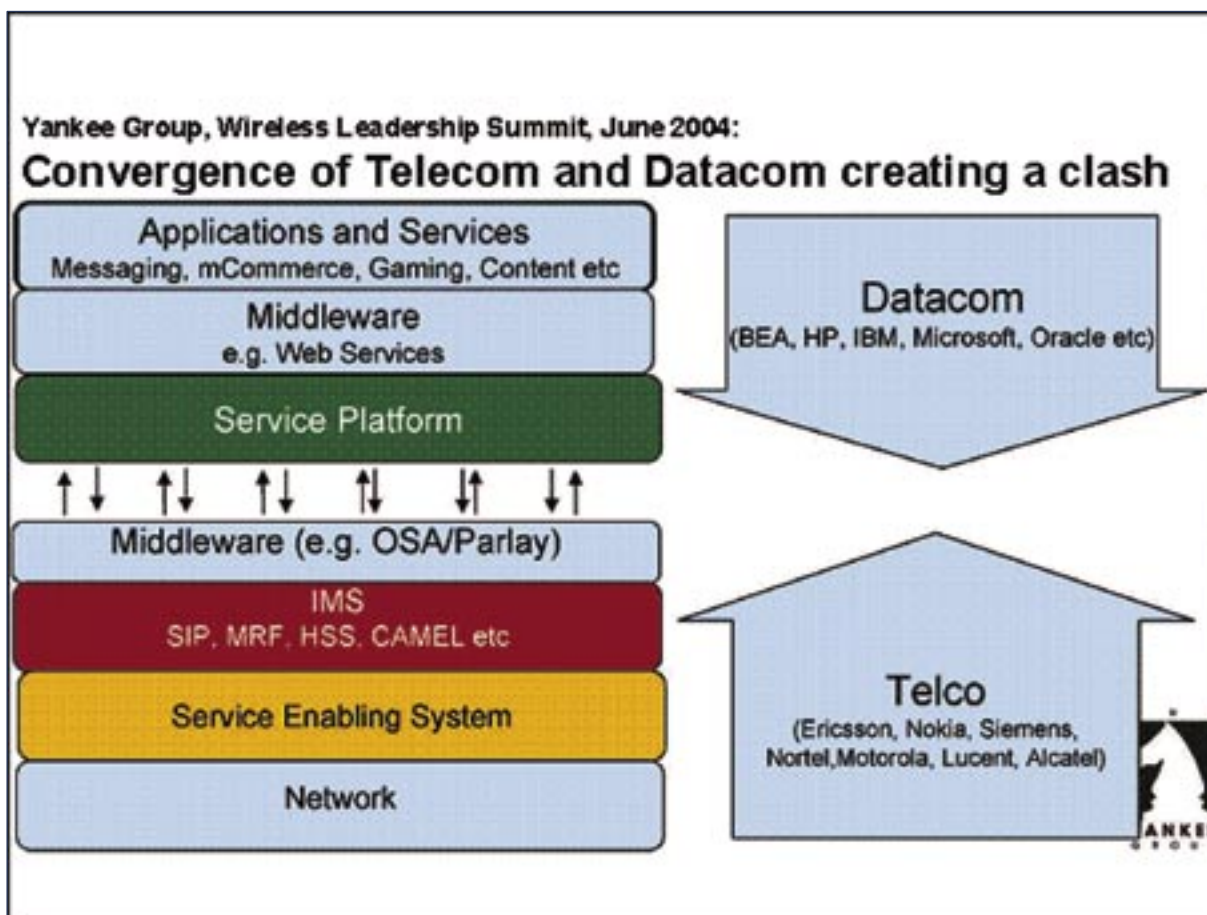
- temeljna logika (*Core Logic*): komponente pružaju temeljne poslovne usluge na različitim razinama (komponenta temeljne logike implementira poslovni proces ili dio njega i koristi druge komponente temeljne logike i/ili konekcije prema resursima);

- veze prema resursima (*Resource Connector*): komponente pružaju (*sintaksnu*) adaptaciju i (*semantičku*) medijaciju prema resursima.

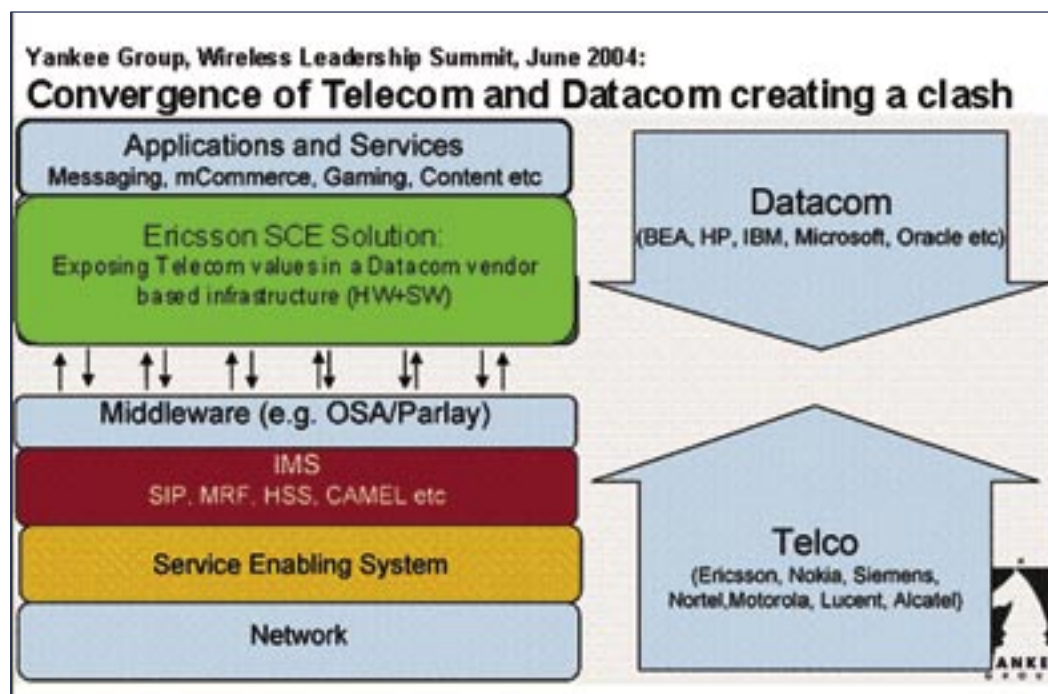
Specifikacija aplikacijske platforme J2EE (Java 2 Enterprise Edition) je primjer specifikacije koja podržava distribuiranu višerednu arhitekturu aplikacijske podrške (*software architecture*). Distribuirana struktura J2EE se sastoji od:

- komponenti klijentskoga reda (*Client-tier*) koje se izvršavaju na klijentovom stroju;
- komponenti *web* reda (*Web-tier*) koje se izvršavaju na J2EE aplikacijskom poslužitelju;
- komponente poslovnoga reda (*Business (EJB)-tier*) koje se izvršavaju na J2EE aplikacijskom poslužitelju;
- red enterprise informacijskih sustava (*Enterprise Information system (EIS)-tier*) koji se izvršava na EIS aplikacijskim poslužiteljima.

Ericssonova SNF preporuka (*Ericsson SNF Recommendation*)



Slika 12. Konvergencija telekoma i datakoma



Slika 13. Pozicija okoline za razvoj usluga u kontekstu telekom i datakom konvergencije

tions) unutar tehnološkog područja - XML Web Services je drugi primjer kod kojega je moguće identificirati četiri logička reda:

- red poslovne fasade (*Business Façade tier*)
- red poslovne jezgre (*Business Core tier*)
- red poslovnih resursa (*Business Resource tier*)
- red infrastrukturnih usluga (*Infrastructure Services tier*); u stvarnosti ovo je kategorija zajedničkih resursa.

4.2.2. Višeslojna arhitektura aplikacijske podrške

Višeslojni pristup je način strukturiranja implementacije aplikacijske podrške na specifične aplikacijske dijelove i generalne dijelove infrastrukturne dijelove/platforme, gdje se platforma/infrastrukturni dio može sastojati od više slojeva (Sl.15).

Da bi izvedba i upravljanje aplikacijama bilo efikasno, važno je odabrati platformu koja podržava jednostavnu integraciju između redova i koja može biti korištena u cijelom rasponu redova s domenom sustava u fokusu. Ovakve familije platformi pružaju okruženje koje je obično bazirano na poznatom modelu komponenti, kao što su J2EE ili .NET.

Važno je za primijetiti da podjela sustava aplikacijske podrške u različite slojeve vrijedi za implementacije sučelja protokola i dijelova aplikacijske podrške.

Potrebne funkcionalnosti u svakom sloju su:

- podrška za prezentacijsku fasadu (*Presentation Façade Support*): portali/grafička sučelja, prikazivanje u različitim kanalima i uređajima, web servisi, prevodilački sustavi za (sintaksnu) adaptaciju i (semantičku) medijaciju prema ra-

zličitim informacijskim i komunikacijskim standardima, pristupnicima koji uključuju upravljanje sigurnošću pristupa, itd.

- podrška za temeljnu logiku (*Core Logic Support*), uz podršku za ostale redove: orkestracijski sustav/sustav koji određuje slijed poslovnih procesa (*orchestration/work flow engine*), sustav za pravila (*rule engine*), podršku za transakcije, sinkrono i asinkrono komuniciranje porukama, itd.
- podrška za spojeve prema resursima (*Resource Connector Support*): prevodilački sustavi za sintaksnu adaptaciju i semantičku medijaciju, itd.

Aplikacijski poslužitelj koji zadovoljava J2EE specifikaciju aplikacijske platforme je primjer platforme koja pruža veliki stupanj podrške samom aplikacijskom sloju. Pregled aplikacijskih poslužitelja dostupnih na J2EE platformi je prikazan na Slici 16.

Od posebne važnosti je podrška za XML web usluge, kao što je specificirano u JSR-109 *Web Services*, JSR-101 *JAX-RPC*, i J2EE arhitektura spojnice (*connector architecture*), specificirano u JSR 112 *JCA*.

4.2.3. Automatizacija poslovnih procesa

Većina kompanija koristi aplikacijsku podršku za automatizaciju sve više svojih poslovnih procesa (*work flows*). Takvi procesi se sastoje od podprocesa/aktivnosti kao komponenti. Kako se poslovanje mijenja s vremenom, nove mogućnosti se otvaraju, druge se zatvaraju, i slično, što dovodi do vitalne potrebe mogućnosti prilagodbe novim zahtjevima poslovanja na brz i efikasan način.

Za postizanje takve funkcionalnosti važno je identificirati i izolirati stabilne dijelove procesa od onih procesa koji su više podložni promjenama. Stabilni dijelovi procesa trebaju biti implementirani kao komponente koje pružaju usluge po definiranim ugovorima. Dijelovi podložniji promjenama moraju biti sposobni mijenjati se uz brzu i jednostavnu kombinaciju komponenti u nove procese.

Jezici, alati i okoline za izvršavanje i orkestraciju poslovnih procesa su već zastupljeni na tržištu. Jezik za izvršavanje poslovnih procesa za web usluge (BPEL4WS - *Business Process Execution Language for Web Services*) je nastao kao zajednička inicijativa kompanija BEA, IBM i Microsoft i u procesu je OASIS WSBPEL TC standardizacije. Druga zajednička inicijativa BEA-e i IBM-a je jezik za izvršavanje poslovnih procesa za Javu (BPELJ - *Business Process Execution Language for Java*) specificiran u zahtjevu za specifikaciju Jave broj 207 (JSR - *Java Specification Request*).

Poslovni procesi mogu biti opisani na dva načina. Izvršni poslovni procesi modeliraju stvarno ponašanje sudionika u poslovnoj interakciji. Poslovni protokoli, naprotiv, koriste opise procesa koji specificiraju ponašanje međusobno vidljivih poruka za svaku stranu uključenu u protokol, bez otkrivanja unutarnjega ponašanja.

Opisi procesa za poslovne protokole se nazivaju apstraktnim procesima. BPEL4WS služi modeliranju ponašanja izvršnih i apstraktnih procesa.

BPEL4WS je jezik za formalnu specifikaciju poslovnih pro-

cesa i protokola poslovnih interakcija. Na taj način BPEL4WS, proširuje interakcijski model web usluga i definira interoperabilni model koji iskorištava proširenja automatizacije poslovnih procesa unutar same korporacije, ali i unutar poslovno-poslovnih (B2B) prostora komunikacije.

4.2.4. Arhitektura bazirana na komponentama

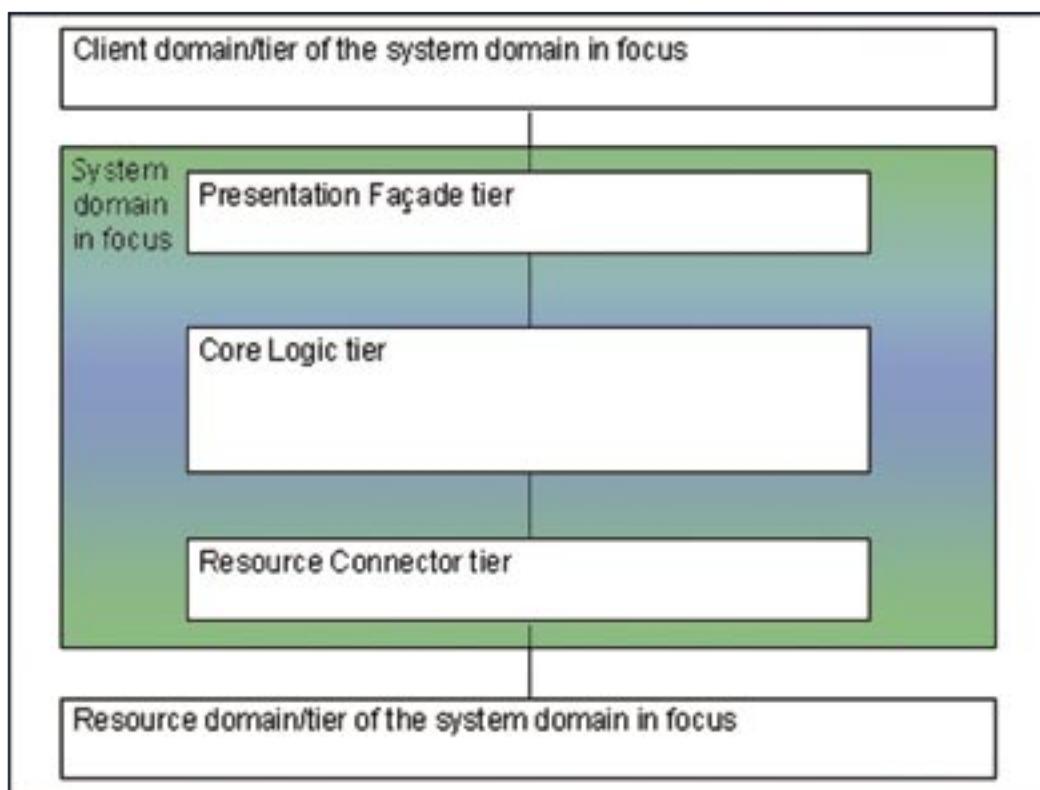
Za brz odgovor na nove zahtjeve postavljene prema aplikacijama programske podrške nije dovoljno samo fokusirati se na orkestraciju poslovnih procesa, već je potrebno identificirati, definirati i razviti komponente koje pružaju usluge za korištenje u poslovnim procesima (*Component Based Architecture*).

Za stvarnu efikasnost u razvoju i izvedbi komponenti, one moraju biti razvijene u skladu s dobro definiranim modelom komponenti. Model komponenti specificira pravila koja same komponente moraju poštovati.

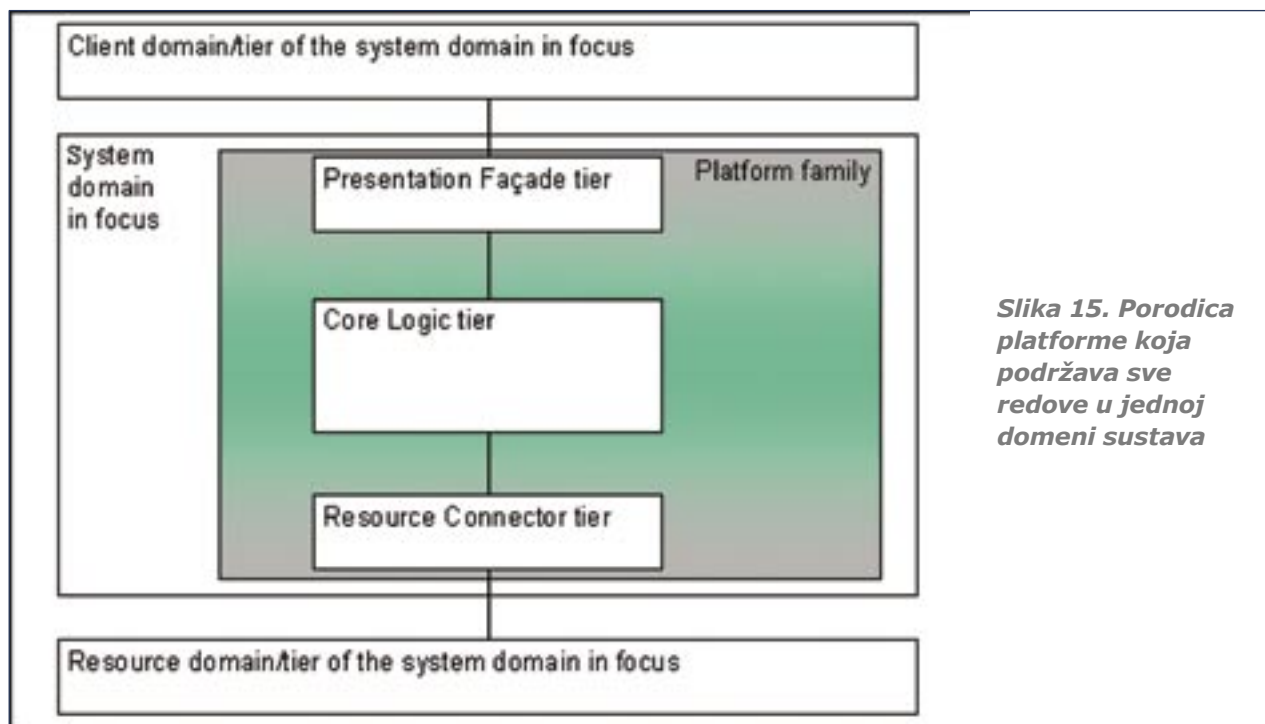
J2EE predstavlja takav model komponenti i kao takva predstavlja okosnicu okoline za razvoj usluga.

4.2.5. Ugovori i uslužno orijentirana arhitektura

Jedan dio modela komponenti predstavlja model ugovora koji specificira pravila koja se moraju poštovati za uspješno izlaganje i korištenje usluge pružene samom komponentom (*Contracts and Service Oriented Architecture*). Takav ugovorni model može biti manje ili više ovisan u odnosu na



Slika 14. Pregled zajedničkog sloja domene aplikacijskih sustava



Slika 15. Porodica platforme koja podržava sve redove u jednoj domeni sustava

okruženje/platformu i podržavati slobodno spajanje (*loose coupling*) između različitih strana zastupljenih ugovorom na različitim razinama.

Arhitektura orijentirana na usluge (SOA - *Service Oriented Architecture*) spada u kategoriju ugovornih modela koji su neovisni o okruženju/platformi same komponente i kao takvi neovisni o implementaciji same komponente.

SOA također stavlja naglasak na transparentnost tako da bi SOA trebala biti primijenjena u svakom slučaju kada postoji potreba za slobodnim spajanjem strana u ugovoru, što je posebno pogodno za spajanja među redovima u distribuiranoj aplikacijskoj podršci.

Okolina za razvoj usluga ima za cilj poboljšavanje interoperabilnosti u interakciji poslovnih subjekata, što nameće potrebu za efikasnim i pouzdanim interoperabilnim tehnologijama baziranim na SOA principu.

Jedna od takvih tehnologija su *web* usluge (WS - *Web Services*) koje je specificirao svjetski *web* konzorcij (W3C - *World Wide Web Consortium*). Proces korištenja *web* usluga između zainteresiranih strana je prikazan na Slici 17.

Web usluge su tehnologija koja je poprimila široku podršku kao važna implementacija uslužno orijentirane arhitekture. Razlog tome su mogućnosti *web* usluga za integraciju ekstremno heterogenih aplikacija preko Interneta primjenom raspodijeljenog računalnog modela. Specifikacija *web* usluga je u potpunosti nezavisna o programskim jezicima, operacijskim sustavima te samim platformama i promovira slobodno spajanje između korisnika i davatelja usluga. Tehnologija je bazirana na otvorenim tehnologijama kao što su (Sl. 18.):

- *eXtensible Markup Language* (XML);
- *Simple Object Access Protocol* (SOAP);
- *Universal Description, Discovery and Integration* (UDDI);
- *Web Services Description Language* (WSDL).

Korištenje otvorenih standarda pruža široku interoperabilnost među rješenjima različitih davatelja. Ovaj princip omogućava kompanijama da implementiraju *web* usluge bez ikakvog znanja o samom potrošaču usluga i obratno. Samim tim omogućava se jednostavna i kratka integracija koja podržava jednostavnu i dinamičnu uspostavu novih poslovnih partnerstava.

Osim osnovne specifikacije *web* usluga razvija se i skupina naprednih WS-* specifikacija kao što su adresiranje *web* usluga (WS-*addressing*), sigurnost *web* usluga (WS-*Security*), pouzdana izmjena poruka (WS-*ReliableMessaging*) zajednički su razvili Microsoft, BEA i IBM, te su ih prosljedili standardizacijskim tijelima (primarno W3C-u i WS-I-u) kako bi se osigurala njihova interoperabilnost).

Skup protokola za *web* usluge (*Web Service Protocols*) je dizajniran kao familija izgradbenih blokova baziranih na modularnoj arhitekturi SOAP-a. Profili specifikacija su izvedeni tako da definiraju ograničenja korištenja i najbolja iskustva za korištenje specifikacija u različitim kombinacijama.

4.2.6. Zajednički informacijski model

Za rješenja u kojima puno aplikacijskih sustava/komponenti međusobno djeluje najjednostavniji način osiguranja inte-

roperabilnosti je kreiranje zajedničkog informacijskog modela (*Shared Information Model*). Informacijski model, ili dijelovi modela, mogu se potom iskoristiti u više uslužnih ugovora. S tim se postiže kreiranje porodice uslužnih ugovora definiranih na istom informacijskom modelu. Ako ne postoji zajednički informacijski model nameće se potreba za kompleksnim translacijama/adaptacijama i medijacijom informacija. Također, ako ne postoji zajednički informacijski model postoji rizik nemogućnosti podrške i automatizacije poslovnih procesa.

Zajednički informacijski model ne znači da su sve informacije dostupne i poznate svim sustavima/komponentama, a to ne implicira niti da su informacije spremjene u jedinstvenom skladištu.

Međutim, tipovi atributa u specifičnoj informacijskoj domeni su obično povezani s općenitim tipovima objekata koji su zajednički za više informacijskih domena. Na primjer, općeniti objekti kao što su "korisnik", "usluga", i slično, imaju različite skupine atributa asocirane s domenama naplate, aktiviranja i osiguranje usluge, ali je vitalno da svaka od informacijskih domena dijeli isti zajednički informacijski model općenitoga tipa objekta.

U nedostatku takvog modela teško je mapirati informacije ili akcije između različitih domena. Preferira se pažljiva identifikacija zajedničke skupine informacija koja se koristi u poslovanju/sustavima i dodjeljivanje vlasnika koji kontroliraju osvježavanje informacija te uslužnih ugovora za manipulaciju informacijom. Osim toga, ključna je mogućnost

jednostavnog i bezbolnog proširenja zajedničkog dijeljenog informacijskog modela.

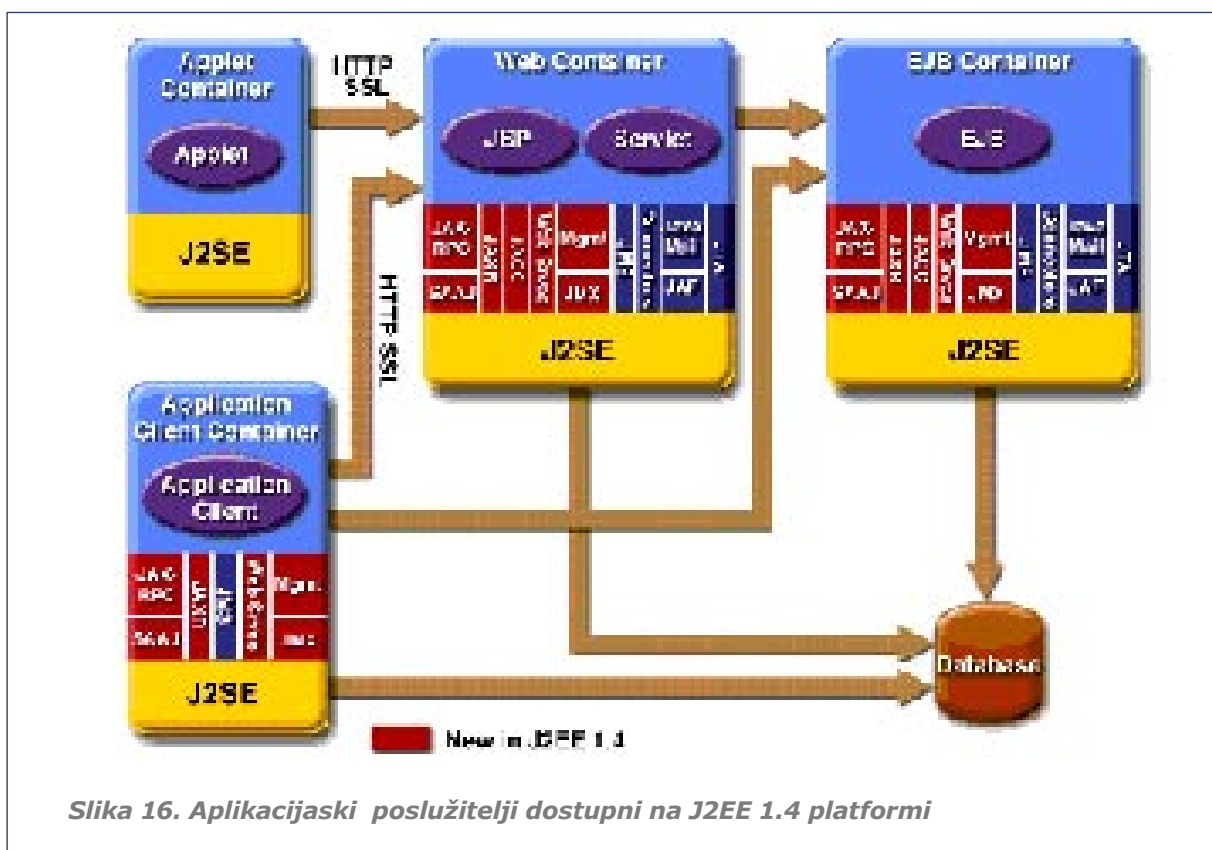
4.2.7. Arhitektura bazirana na usmjeravanju

Prilikom isporuke informacija i komunikacijskih usluga postavljaju se određena pravila usmjeravanja (*Policy Enabled Architecture*) definirana ispunjavanju različitih poslovnih, sustavnih ili drugih ciljeva.

Ukoliko su pravila usmjeravanja za aplikacije, koje isporučuju usluge, definirana unaprijed i neće se nikad mijenjati, tada možemo kreirati aplikacijsku podršku koje je nepromjenjiva (*hard-coded*).

Međutim, to nije slučaj i nameće se potreba mogućnosti postavljanja i primjene usmjeravanja tijekom samog izvođenja. U arhitekturi baziranoj na usmjeravanju obrazac "razdvajanja interesa" je još jednom primijenjen, a u lancu isporuke identificirane su dvije uloge: točka provođenja usmjeravanja (*PEP - Policy Enforcement Point*) i točka donošenja odluke o usmjeravanju (*PDP - Policy decision Point*).

Uloga točke donošenja odluke o usmjeravanju je dodijeljena središnjem aplikacijskom sustavu gdje se usmjeravanja mogu postavljati i gdje se logika za donošenje odluka izvršava po zahtjevu aplikacijskih sustava kojima je pridijeljena uloga točke za provođenje usmjeravanja. Uloga točke za provođenje usmjeravanja je pridijeljena aplikacijskom sustavu koji ima ulogu provjere i provođenja određenih usmjeravanja u specifičnim točkama lanca isporuke usluga.



Jedna od kategorija usmjeravanja je sigurnosna politika. Prilikom automatizacije poslovnih procesa i otvaranja ka B2B interoperabilnosti s različitim razinama povjerenja, važno je postojanje postojećeg sigurnosnog mehanizma. Postojanost mehanizma se ogleda u mogućnosti ispunjavanja sigurnosnih potreba (i odgovaranja na pitanja privatnosti) svih zainteresiranih strana, a ne podrazumijeva, kako bi se na prvi pogled moglo pretpostaviti, da se jednom uspostavljeni odnosi ne mogu mijenjati. Naprotiv, mora postojati mogućnost brzog i fleksibilnog modela upravljanja sigurnosnom politikom. Upravljanje sigurnosnom politikom je svojstvo PDP-a. Sigurnosni PEP je postavljen na točke presretanja u lancu isporuke, obično u graničnim pristupnicima. Druge kategorije politika odnose se na ugovore o razini usluge i politike naplate.

4.2.8. Okolina za razvoj usluga i telekom tehnologije

U prijašnjim poglavljima razjašnjeni su odnosi okoline za kreiranje usluga u kontekstu krosindustrijskih aplikacijskih pristupa i tehnologija koje postaju sve primjenjivije u telekom domeni. Ovo poglavlje će objasniti aspekte specifične za telekom u kontekstu samog mjesta obitavanja okoline za ra-

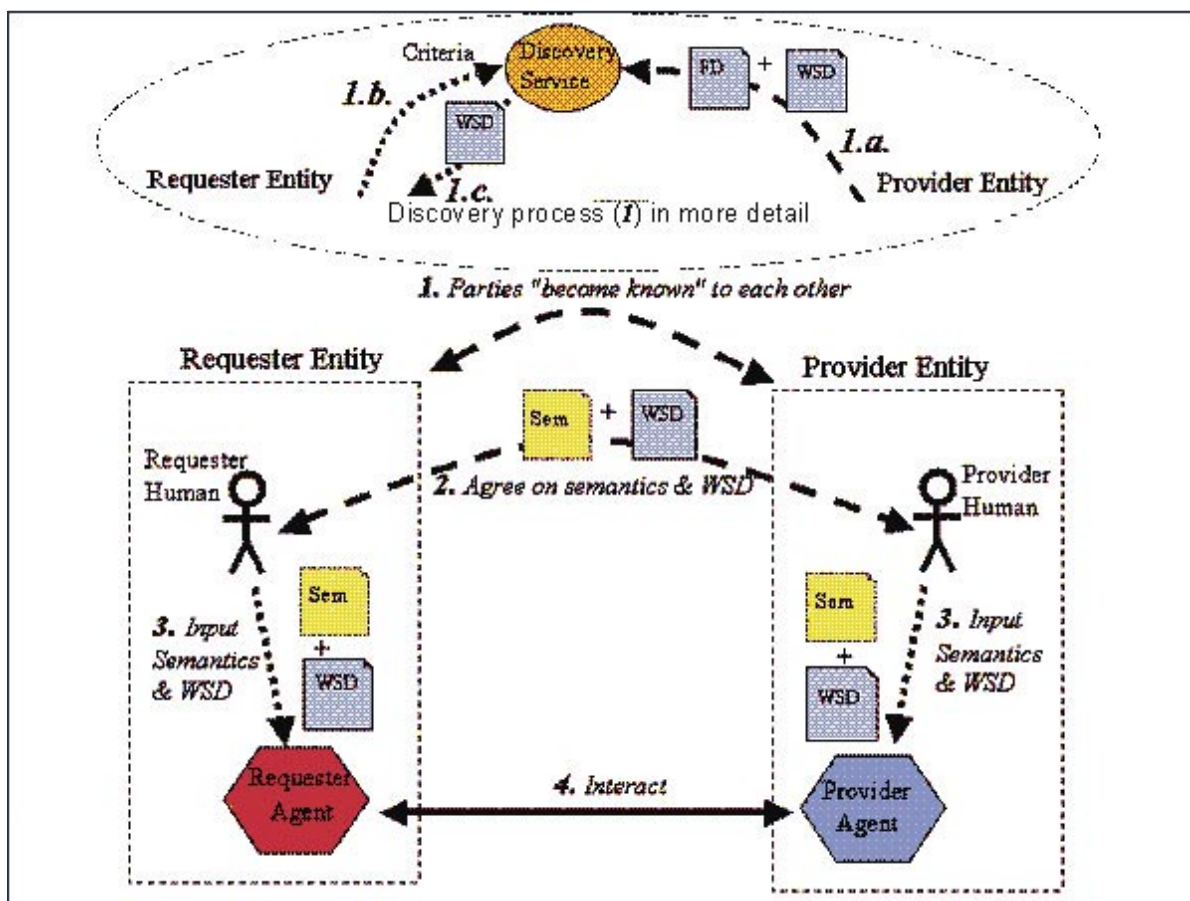
zvoj usluga (SCE and telecom technologies).

Trenutačno postoji veliki broj pristupnih i jezgrenih mrežnih tehnologija. Pojava novih i nestajanje postojećih tehnologija je prirodan proces usmjeren na trend kreiranja jednostavnijih tehnologija i omogućavanje konvergencije različitih mrežnih tehnologija. Ovo je posebno važno za tehnologije jezgrene mreže koje, između ostalog, imaju ulogu spajanja različitih pristupnih tehnologija.

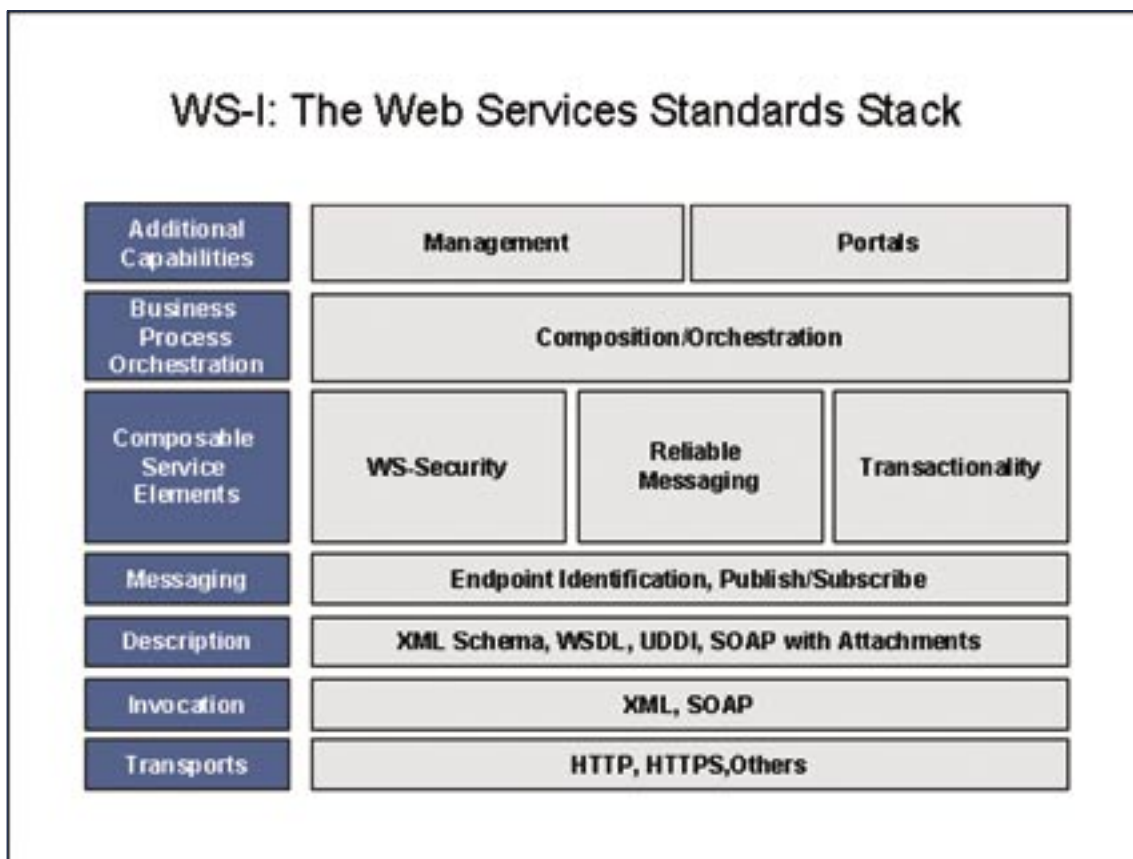
Ono u što možemo biti sigurni je mješavina različitih generacija telekom tehnologija u prijelaznom periodu i potreba za postizanje interoperabilnosti između različitih mrežnih tehnologija (PS i CS mreže, mobilne i fiksne mreže, mobilne mreže druge i treće generacije - 2G, 2,5G, 3G, i slično).

4.2.9. Trendovi telekom tehnologija iz aplikacijske perspective

Za jednostavnu komunikaciju između pojedinaca (P2P - *Person to Person*) konvergencija i/ili interoperabilnost je riješena u jezgrenoj mreži, međutim okolina za razvoj usluga primarno je usredotočena na komunikaciju osoba prema sadržaju/aplikaciji (P2C&A - *Person to Content & Application*) i obratno, komunikaciju aplikacija/sadržaja prema -



Slika 17. Proces korištenja web usluga



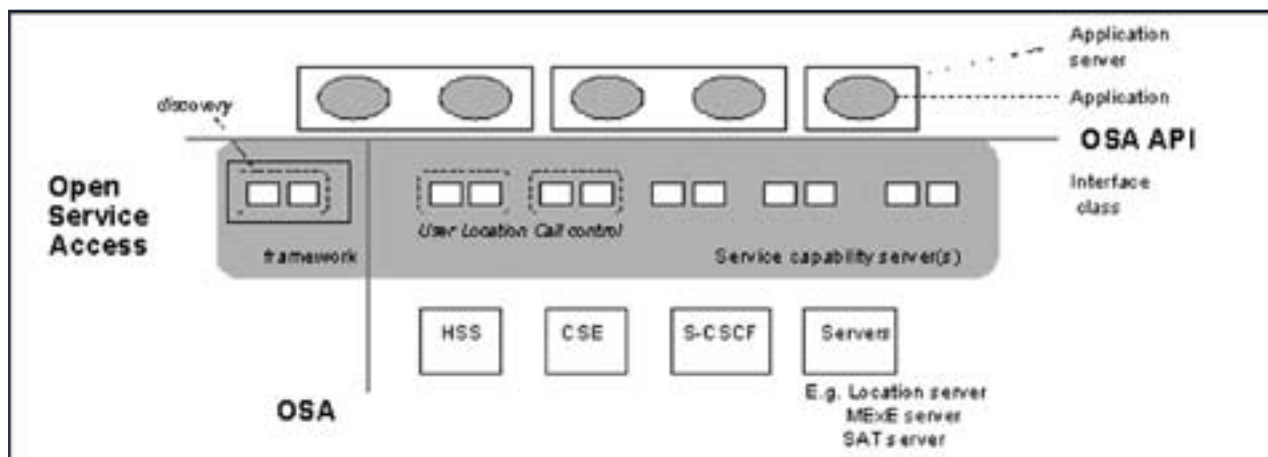
Slika 18. Standardi za web usluge

osobi (C&A2P - Content & Application to Person).

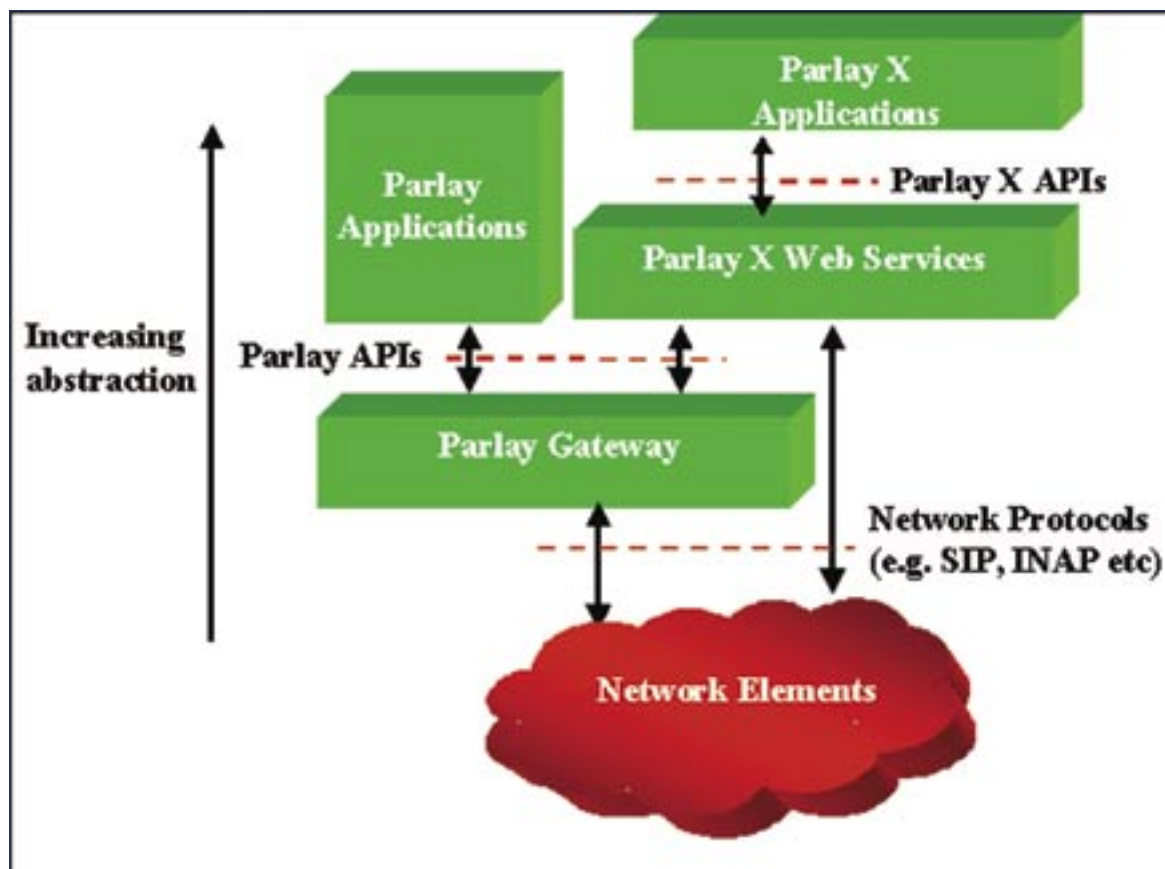
Oslobađanje razvijatelja aplikacija od potrebe za telekom specifičnim znanjima i tehnologijama i izbjegavanje ponovnog razvoja postojećih aplikacija za različite telekom tehnologije nametnulo je potrebu za pristupnicima i medijatorima koji se pouzdaju na IP tehnologije i izlažu mrežne sposobnosti na tehnološki neutralni način. Jedna od takvih inicijativa je otvoreni uslužni pristup - OSA sučelje (Open Service Access) i okruženje definirano Parlay grupom. 3GPP je također uključen u razvoj OSA specifikacija koje je i sam prihvatio. Primjer na Slici 19. prikazuje poje-

dnostavljene ilustracije iz 3GPP TS 23.198 i specificira OSA sučelja i okružje.

Inicijalna sučelja koja je definirala Parlay grupa su bazirana na CORBA tehnologiji. S vremenom Parlay grupa je prihvatila web usluge kao tehnologiju i objavila niz specifikacija baziranih na web uslugama pod imenom Parlay X (SL20). Parlay X sučelja za cilj imaju olakšavanje razvoja aplikacija (Parlay X je također prihvatila 3GPP organizacija). Parlay X je također i preferirana tehnologija za otkrivanje mrežnih mogućnosti trećim stranama, u ovom slučaju preko sigurnosnih pristupnika.



Slika 19. 3GPP prikaz OSA okruženja i sučelja



Slika 20. Parlay X sučelja i arhitektura

Druge inicijative koje usmjerene na aplikacije i mehanizme njihovoga kreiranja (*enablers*) spominje organizacija *Open Mobile Alliance* (OMA). OMA arhitektura je zasnovana na sljedećim principima:

- Princip #1 (*Principle #1*) - kreiranje specifikacija aplikacijskih enablera neovisnih o postojećim mrežama, operacijskim sustavima i programskim jezicima;
- Princip #2 (*Principle #2*) - iskorištavanje postojećih standarda;
- Princip #3 (*Principle #3*) - pružanje otvorenih, interoperabilnih, skalabilnih, proširivih, modularnih aplikacija i specifikacija;
- Princip #4 (*Principle #4*) - omogućavanje prilagodbe usluge u ovisnosti o sposobnosti korisničkog uređaja, mrežnim karakteristikama i korisničkim preferencijama;
- Princip #5 (*Principle #5*) - pružanje sigurnosti usklađene s modelima sigurnosti na internetu i zaštita korisničke privatnosti.

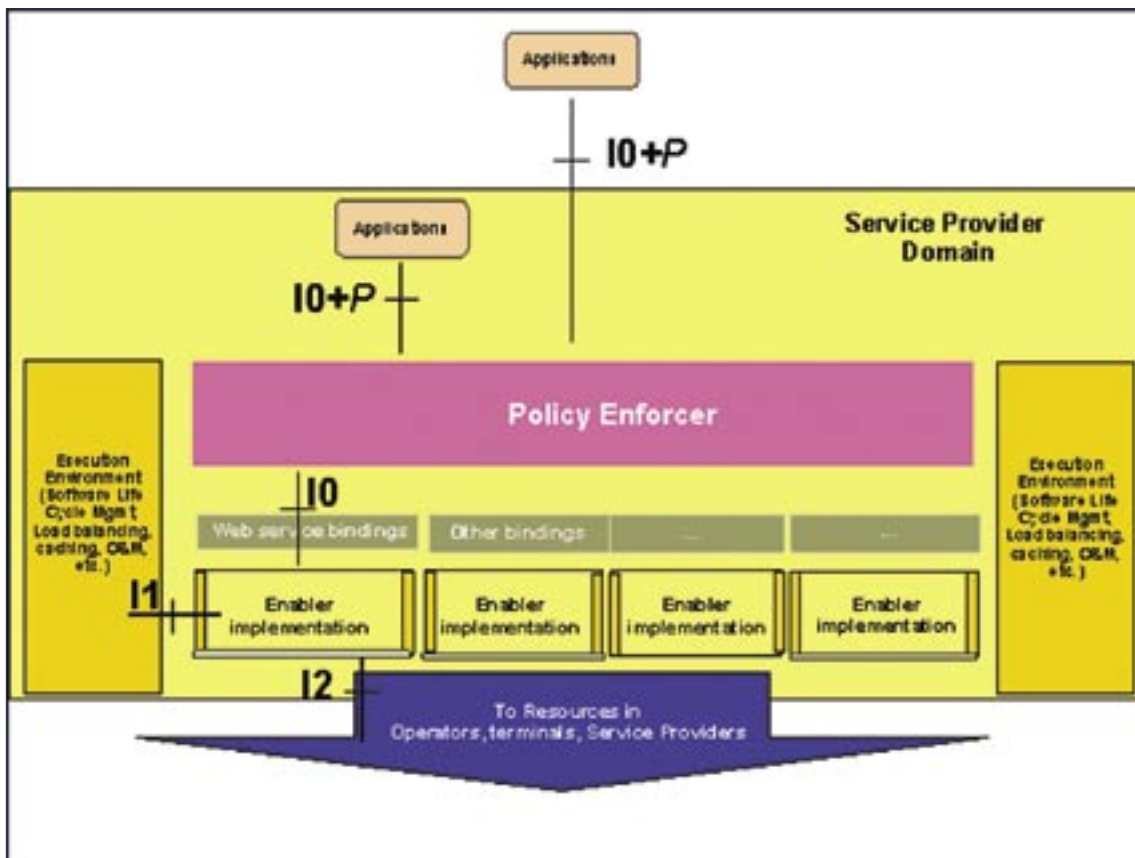
OMA je također skicirala OMA Uslužno okružje (OSE - *OMA Service Environment*), kako je prikazano na Slici 21. Takvo rješenje je primjenjivo za arhitekturu okruženja za kreiranje usluga u operatorovoj poslovnoj domeni (tj. Domeni davatelja kanala), što je na Slici 21. označeno kao domena davatelja usluga (*Service Provider Domain*).

Također, okolina za razvoj usluga je prvenstveno namijenjena stvaranju preduvjeta da mrežni operatori uspostavljaju partnerstva s većim brojem suradnika (*mass partnering*) što nameće pitanje privatnosti korisničkih podataka koji su definirani u *Liberty Alliance Project* (LAP) standardima. Pregled LAP standarda je prikazan na Slici 22.

4.2.10. Tehnološki trendovi iz perspective enablera

Parlay/OSA, OMA i slične inicijative očito su usmjerene na olakšavanje razvoja aplikacija. Međutim, postavlja se pitanje kakvi su trendovi u jezgrenom dijelu mreže i na koji način oni utječu na aplikacije i pristup njima. Odgovor na ovo pitanje nalazimo u činjenici da se temeljni dijelovi telekom infrastrukture sve više, ako ne i u potpunosti, oslanjaju na IP tehnologije. Glavna inicijativa u jezgrenom mreži odnosi se na višemedijski podsustav zasnovan na Internet protokolu (IMS - *IP Multimedia Subsystem*) kojega je definirala organizacija 3GPP (TS 23.228). IMS se fokusira na višemedijsku kontrolu poziva baziranu na sesiji i protokolu za uspostavu sesije (SIP - *Session Initiation Protocol*).

Glavna zadaća IMS-a je višemedijska kontrola sesije osoba-osoba (P2P), koju podržava aplikacija u korisničkom uređaju (UE - *User Equipment*), međutim, s aspekta okoline za



Slika 21. OMA uslužno okruženje OMA Service Environment - konceptualna arhitektura i kategorije sučelja

razvoj usluga je činjenica da 3GPP definira IMS uslužno/aplikacijsko okruženje za aplikacije udomljene na aplikacijskim poslužiteljima (AS - Application Servers).

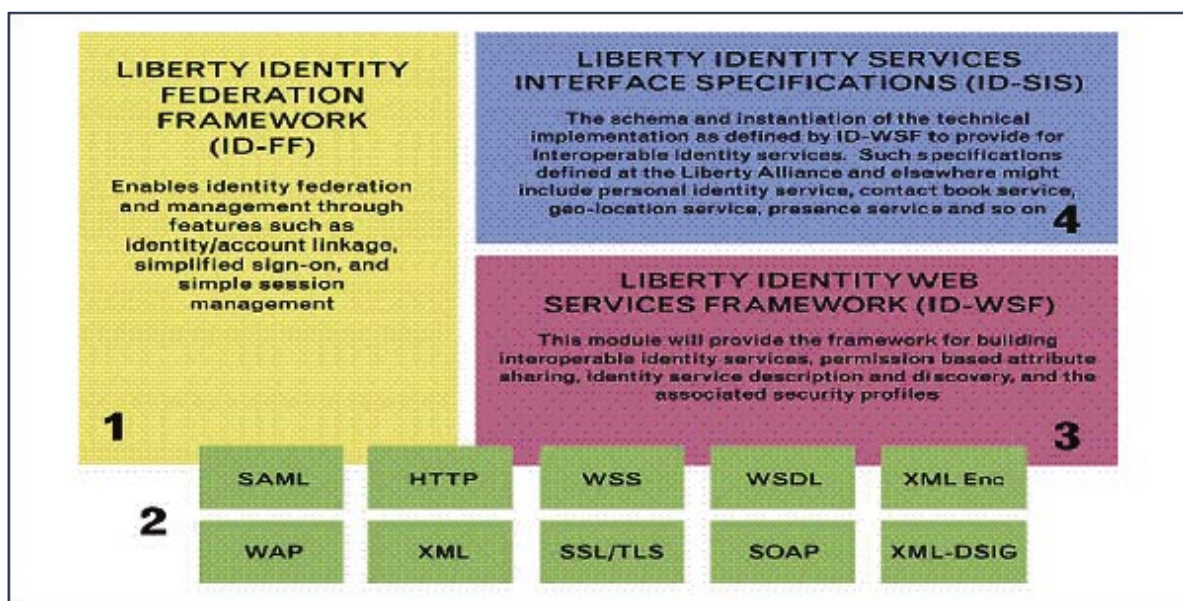
Aplikacije komuniciraju s jezgrom IMS podsustava preko:

- ISC (IMS Service Control) sučelja (baziranoga na SIP protokolu) prema funkciji S-CSCF (Serving Call/Session Control Function)
- Si/Sh sučelja (baziranog na Diameter protokolu) prema kućnom pretplatničkom poslužitelju (HSS - Home Subscriber Server)

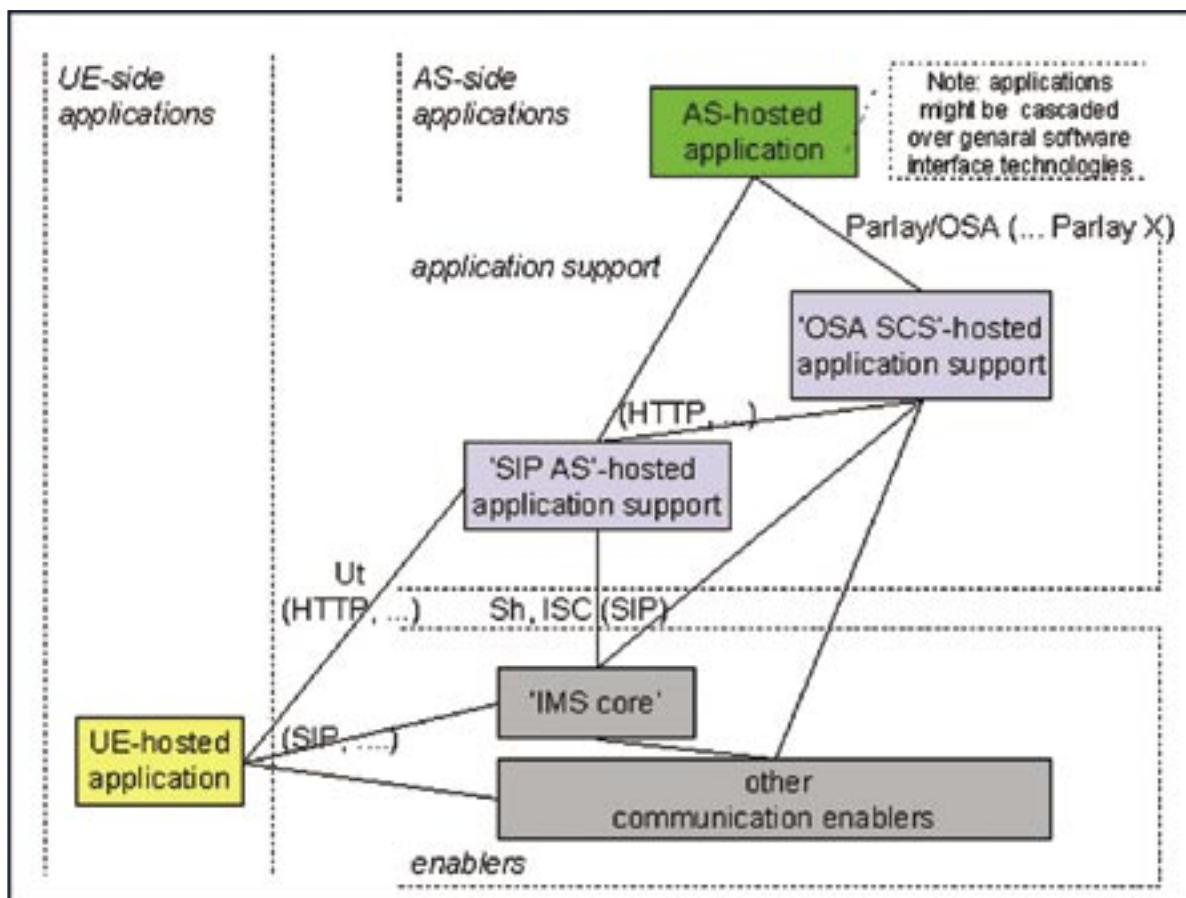
3GPP IMS definira tri različite kategorije IMS uslužnih/aplikacijskih okruženja:

- Funkcija IM SSF (IP Multimedia Switching Service Function) - tradicionalne aplikacije (Legacy Applications)
- SIP aplikacijski poslužitelj (SIPAS) - izvorne IMS aplikacije
- Poslužitelj OSA SCS (OSA Service Capability Server) - uslužne mogućnosti izložene preko OSA okruženja prema OSA aplikacijskim poslužiteljima.

Podrška aplikacijama, kao što su prisutnost i upravljanje ko-



Slika 22. Pregled standarda koje propisuje LAP organizacija



Slika 22. Aplikacije, podrška aplikacijama i komunikacijski enableri

risničkim grupama, je specificirana za udumljavanje u SIP aplikacijskim poslužiteljima. Ovo su primjeri zajedničkih karakteristika koje upotpunjuju višemedijsku sesiju preko protokola SIP i njegovih proširenja. Uglavnom su usmjerene na podršku aplikacija na korisnikovom terminalnom uređaju, ali isto tako aplikacije udumljene na SIP aplikacijskom poslužitelju mogu konzumirati iste sposobnosti.

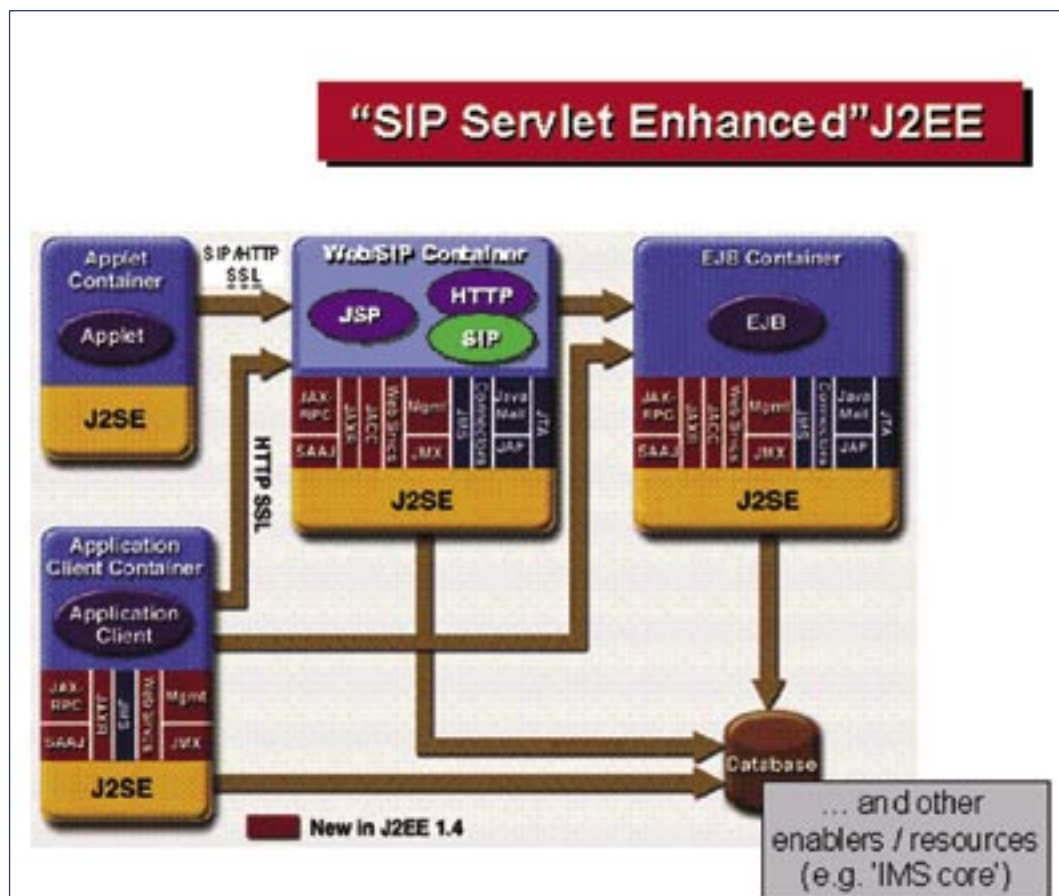
SIP bazirana podrška za aplikacije nije jedina sposobnost potrebna u aplikacijama. Za korisničke aplikacije postoje i druga sučelja i protokoli, kao što je sučelje IMS Ut (XCAP i HTTP bazirano) za manipulaciju podacima u aplikacijskoj podršci udumljenoj u SIP aplikacijskim poslužiteljima.

Za aplikacije koje su udumljene na aplikacijskim poslužiteljima i koje su u fokusu okoline za kreiranje usluga, nameće se potreba za sučelja koja nadmašuju mogućnosti SIP protokola (i njegovih proširenja), a isto tako i karakteristika koje osigurava IMS podsustav. Na primjer, sučelje *Parlay/OSA* pruža veći raspon sposobnosti podrške aplikacijama za aplikacije udumljene u aplikacijskim poslužiteljima i otvara nove mogućnosti konvergencije telekom tehnologija sa stajališta aplikacije udumljene na aplikacijskom poslužitelju. Također, da ponovimo sučelje *Parlay/OSA* omogućava izlaganje telekom sposobnosti trećim stranama putem *web* usluga i sigurnosnih pristupnika te donosi mogućnosti kontrole i provođenja ugovora o razini usluge.

Preferirana tehnologija okruženja za razvoj usluga je tehnologija J2EE. Ovdje se nameće potreba za razmatranjem i uključivanjem SIP aplikacijskog poslužitelja u platformu J2EE. Podrška tehnologijama utemeljenim na protokolu SIP je implementirana kao servlet. Servleti su izvedeni u SIP servlet kontejneru koji podržava i SIP i HTTP komunikaciju, kako je i prikazano na *Slici 23*.

4.2.11. Okolina za razvoj usluga u kontekstu telekom tehnologija i aplikacijske podrške

Ukoliko kombiniramo pristup izlaganja mrežnih sposobnosti s krosindustrijskim aplikacijskim okolinama za razvoj aplikacija i B2B interoperabilnošću dobijamo situaciju prikazanu na *Slici 23*. (con= spojnik ka *enablerima/resursima*; SEG = sigurnosni pristupnik). Ovaj kontekst je osnova okoline za razvoj usluga uz pomoć koje su usluge omogućene aplikacijama i pripadajuće administrativne usluge koje pruža operator (koji djeluje kao davatelj kanala) stavljene na raspolaganje davateljima usluga dodatne vrijednosti (VASP - *Value Added Services Providers*) preko *web* usluga i sigurnosnih pristupnika tako da omogućavaju efikasan razvoj aplikacija, pri tom ne pokazujući stvarnu



Slika 22. Tehnologija J2EE proširena SIP servletom

kompleksnost telekomunikacijskih tehnologija. Ukoliko se usredotočimo na postojeću okolinu za kreiranje usluga i pogledamo ka budućnosti, rješenje koje uspješno inkorporira dosadašnje tehnologije i nadolazeće tehnologije u zajedničkoj okolini, dobivamo arhitekturu, prikazanu na Slici 24., koja se sa svojim sposobnostima i otvorenosću može uspješno nositi sa zadaćom razvoja i isporuke konvergentnih i kombinatornih usluga sadašnjih i budućih generacija mreža te upravljanjem njima.

5. Zaključak

Kao zaključak navedimo osobine koje suvremena okolina za kreiranje usluga danas i u budućnosti mora omogućavati svim protagonistima, bez obzira na njihove poslovne uloge, u procesu kreiranja, isporuke i konzumiranja usluga.

1. Kratko vrijeme proteklo od ideje do isporuke usluge krajnjim korisnicima)

- Operator (davatelj kanala) mora pružiti široki spektar aplikacijskih usluga, putem efikasnoga okruženja za razvoj, širokoj zajednici razvijatelja aplikacija s limitiranim znanjem tehnologija specifičnih za telekom industriju.
- Razvijatelj aplikacija mora pronaći i koristiti potrebne usluge za aplikacije te na taj način brzo implementirati nove aplikacije.

- Operator (davatelj kanala) mora omogućiti administrativne usluge za operacije na ponuđenim aplikacijskim uslugama vanjskim isporučiteljima aplikacija i sakupljačima sadržaja i aplikacija koji imaju limitirano znanje tehnologija specifičnih za telekom industriju, putem jednostavnih i intuitivnih portala.

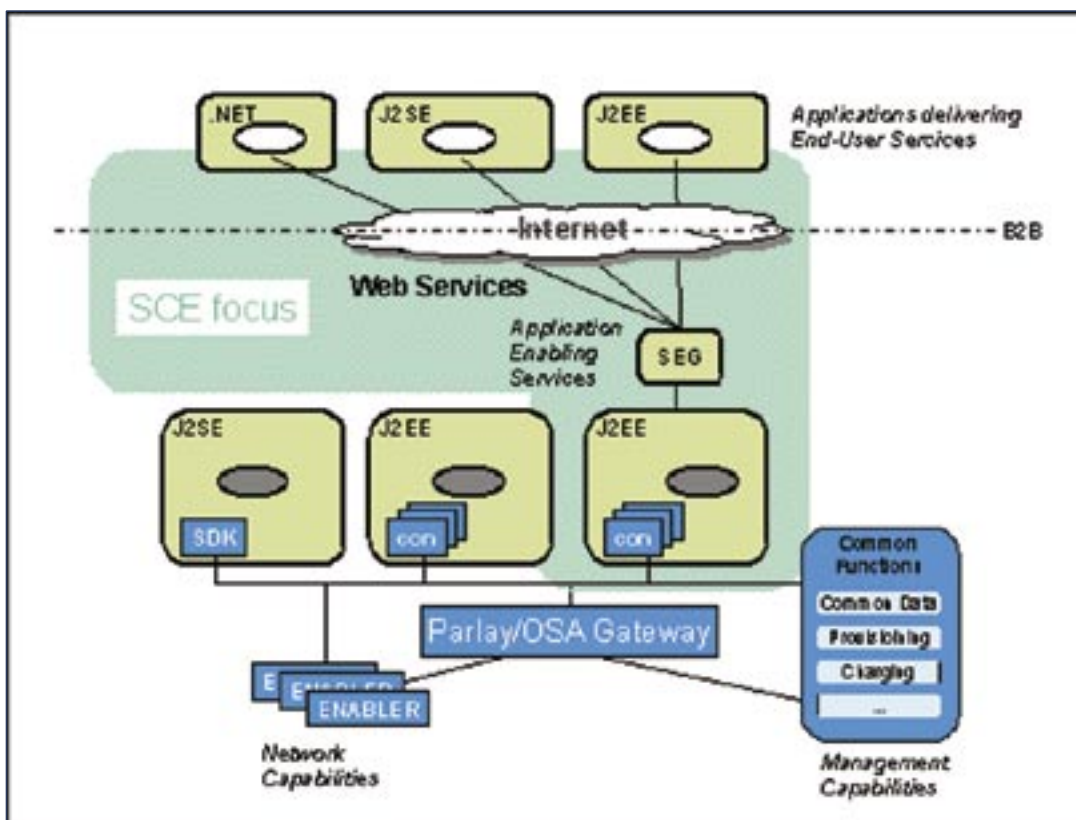
- Operator (davatelj kanala) mora, putem jednostavnih i intuitivnih portala osigurati marketinške kanale za usluge vanjskih isporučitelja aplikacija te sakupljača sadržaja i aplikacija, koji imaju limitirano znanje o tehnologijama specifičnim za telekom industriju.

- Isporučitelj aplikacija će imati na raspolaganju veći broj aplikacija koje može udomiti te ih ponuditi sakupljačima sadržaja i aplikacija. Nadalje, isporučitelj aplikacija mora od operatora dobiti na raspolaganje i administrativne usluge.

- Sakupljač sadržaja i aplikacija dobija na raspolaganje veći broj usluga koje može ponuditi krajnjim korisnicima. Nadalje on dobija i kanal za reklamiranje putem portala za registraciju usluga.

2. Podrška za fleksibilne modele naplate usluga

Zahtjevi korisnika nameću potrebu za različitim modelima naplate za različite usluge, kao što je naplata na bazi sadržaja, prometa, primjena podjele dobiti s partnerom i slično.



Slika 23. Tipično rješenje okoline za razvoj usluga u kontekstu krosindustrijske aplikacijske okoline i telekom okoline

- Operator (davatelj kanala) dobija mogućnost upravljanja naplatom aplikacijskih usluga na fleksibilan način.
- Operator (davatelj kanala) može ponuditi usluge naplate kao dio usluga za omogućavanje aplikacija. Razvijatelj aplikacije je u mogućnosti razvijati aplikacije bez razmatranja načina i modela naplate (u ovom slučaju aplikacijsko okruženje podržava indentifikaciju isporučiteljskih usluga);
i/ili
- Razvijatelj aplikacija je u mogućnosti koristiti usluge za omogućavanje aplikacija za naplatu bez poznavanja pravila za naplatu usluga s tim da izvršenje usluge mora biti reportirano aplikaciji za naplatu;
i/ili
- Razvijatelj aplikacija je u mogućnosti koristiti usluge za naplatu bez poznavanja pravila za naplatu, ali mora voditi računa o raspoloživosti sredstava.
- Isporučitelj aplikacija i sakupljač sadržaja i aplikacija koristi usluge naplate za usluge na fleksibilan način.

3. Troškovi poslovanja su minimizirani za svaku uslugu

S porastom broja ponuđenih usluga ukupna cijena životnih

ciklusa usluga raste, pa troškovi postaju sve važniji u svakodnevnom poslovanju.

- Operator može upravljati svojom ponudom prema razvijateljima aplikacija, isporučiteljima aplikacija i sakupljačima sadržaja i aplikacija na efikasan način, što doprinosi smanjenju troškova.
- Operator (davatelj kanala) može pružiti administrativne usluge za korištenje usluga putem portala koje koriste ostali zainteresirani u lancu ponude.
- Operator (davatelj kanala) može ponuditi marketinški kanal putem samoposlužujućih portala ka krajnjim korisnicima.

4. Dobra podrška interoperabilnosti s poslovnim partnerima

Sve više partnera naglašava potrebu za boljim B2B sučeljima baziranim na standardnim tehnologijama kao što su web usluge.

- Operator (davatelj kanala) može pružiti usluge velikom broju razvijatelja aplikacija, pružitelja aplikacija te sakupljača sadržaja i aplikacija putem standardnih industrijskih sučelja, kao što su web usluge.

• Razvijatelji i isporučitelji aplikacija te sakupljači sadržaja i aplikacija mogu konzumirati usluge koje nude operatori (davatelj kanala), putem B2B sučelja baziranih na industrijskim standardima kao što su web usluge.

5. Podrška marketinškoj strategiji orijentiranoj na krajnjega korisnika

Načini marketiranja moraju biti neovisni o pristupu i korisničkom uređaju.

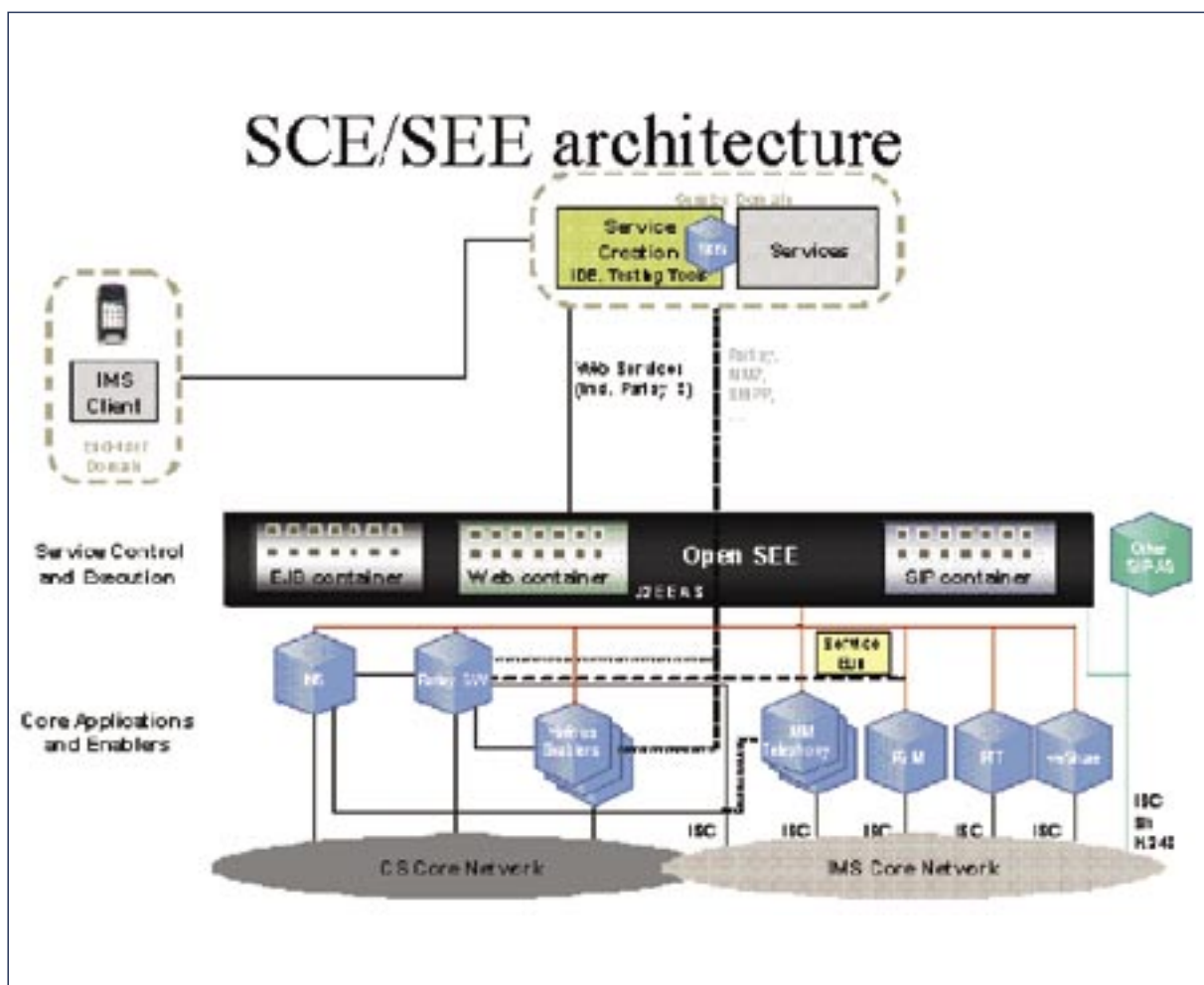
- Operator (davatelj kanala) može pružati usluge aplikacijama koje ili skrivaju tip pristupne mreže i tip korisničkog uređaja ili, ukoliko nije oportuno da ih sakrije, izlažu tip pristupne mreže i korisničkog uređaja. U prvom slučaju radi se o uključivanju u aplikaciju adaptacije za određeni tip pristupne mreže i uređaja. U drugom slučaju radi se o dijeljenju atributa i sposobnosti korisnika radi postizanja cilja: isporuke usluge, neovisno o tipu mreže i uređaja.
- Razvijatelj aplikacija, isporučitelj aplikacija i skupljač sa-

držaja i aplikacija je u mogućnosti razvoja/pružanja usluga za krajnjeg korisnika neovisno o tipu pristupne mreže i uređaja, bilo uz pomoć adaptacija od strane operatora (kao davatelj kanala) ili korištenjem usluga koje omogućuju takve adaptacije.

6. Razvoj usluga neovisno o dobavljačima mrežne opreme

Različitosti u mrežnoj opremi različitih proizvođača ostaju skrivene razvijateljima aplikacija.

- Operator (davatelj kanala) može pružiti usluge za aplikacije skrivajući mrežnu opremu i njene razlike.
- Razvijatelj aplikacija može razvijati aplikacije bez znanja o specifičnostima mrežne opreme potrebne za cjelovitu komunikaciju.
- Isporučitelj aplikacija može izvesti aplikacije bez znanja o specifičnosti mrežnih elemenata potrebnih za cjelovitu komunikaciju.



Slika 24. Konvergentna okolina za kreiranje usluga

6. Popis kratica

AD	- Application Developer	SCS	- Service Capability Server
AP	- Application Provider	S-CSCF	- Serving Call/Session Control Function
AS	- Application Server	SDP	- Service Delivery Platform
B2B	- Business to Business	SIP	- Session Initiation Protocol
BPEL	- Business Process Execution Language	SLA	- Service Level Agreements
BPELJ	- Business Process Execution Language	SMS	- Short Message Service
BPEL4WS	- Business Process Execution Language For Web Services	SNF	- Service Network Framework
CAA	- Content and Application Aggregator	SOA	- Service Oriented Architecture
C&A2P	- Content And Application To Person	SOAP	- Simple Object Access Protocol
CORBA	- Common Object Request Broker Architecture	TINA-C	- Telecommunications Information Networking Architecture Consortium
CP	- Content Provider	TTM	- Time To Market
EIS	- Enterprise Information System	UDDI	- Universal Description, Discovery and Integration
EJB	- Enterprise Java Beans	VAS	- Value Added Service
eTOM	- Enhanced Telecom Operations Map	VASP	- Value Added Services Provider
EU	- End User	W3C	- World Wide Web Consortium
3GPP	- 3rd Generation Partnership Project	WSDL	- Web Services Description Language
HSS	- Home Subscriber Server	WS-I	- Web Services Interoperability
HTTP	- Hyper Text Transfer Protocol	XCAP	- XML Configuration Access Protocol
IMS	- IP Multimedia Subsystem	XML	- Extensible Markup Language
IM SSF	- IP Multimedia Service Switching Function		
ISC	- IMS Service Control		
ISV	- Independent Software Vendor		
JCA	- Java Connector Architecture		
JSR	- Java Specification Request		
J2EE	- Java 2 Enterprise Edition		
J2SE	- Java 2 Standard Edition		
LAP	- Liberty Alliance Project		
MMS	- Multimedia Messaging System		
NP	- Network Provider		
OASIS	- Organization for the Advancement of Structured Information Standards		
OMA	- Open Mobile Alliance		
OSA	- Open Service Access		
OSA	- Open Services Architecture		
PDP	- Policy Decision Point		
PEP	- Policy Enforcement Point		
P2P	- Person to Person		
P2C&A	- Person To Content And Application		
SCE	- Service Creation Environment		

7. Literatura

- [1] Ericsson interna literatura
 [2] TM Forum NGOSS eTOM (GB921)
<http://www.tmforum.org/browse.asp?catID=1647>
 [3] TM Forum NGOSS Technology-Neutral Architecture
<http://www.tmforum.org/browse.asp?catID=1685>
 [4] TM Forum NGOSS Shared Information/Data Model
<http://www.tmforum.org/browse.asp?catID=1684>
 [5] J2EE
<http://java.sun.com/j2ee/1.4/docs/index.html>
<http://jcp.org/en/jsr/detail?id=151>
<http://jcp.org/en/jsr/tech?listBy=3&listByType=platform>
 [6] Web Services
 - W3C <http://www.w3.org/2002/ws/>
 - WS-I <http://www.ws-i.org/>
 [7] SOA / Web Services
 - BEA <http://dev2dev.bea.com/technologies/soa/index.jsp>
 - IBM <http://www-306.ibm.com/software/solutions/webservices/>
 - MSFT <http://msdn.microsoft.com/webservices/>,
<http://msdn.microsoft.com/library/en-us/dnanchor/html/WebServicesAnchor.asp>
 - Oracle <http://www.oracle.com/technology/tech/webservices/index.html>
 - Sun <http://java.sun.com/webservices/index.jsp>
 - OMA http://www.openmobilealliance.org/release_

[program/owser_v10.html](#)

- ZapThink <http://www.zapthink.com/index.html>

[8] Parlay/OSA specifications

- <http://www.parlay.org/specs/index.asp>

- <http://www.3gpp.org/specs/specs.htm>

TS 23.127, TS 23.198, TS 29.198-xx, TS 29.199-yy, TS 29.998-zz

[9] OMA (Open Mobile Alliance)

http://www.openmobilealliance.org/release_program/index.html

(see also http://member.openmobilealliance.org/ftp/Public_documents/ARCH/Permanent_documents/)

OMA Architecture Requirements

http://www.openmobilealliance.org/release_program/docs/RD/OMA-RD_Architecture_V1_0-20031021-A.pdf

OMA Service Environment (OSE)

http://www.openmobilealliance.org/release_program/docs/RD/OMA-Service-Environment-V1_0-20040907-A.pdf

[10] LAP (Liberty Alliance Project) Specifications

<http://www.projectliberty.org/resources/specifications.php>

[11] BPEL4WS

- BEA <http://dev2dev.bea.com/techtracks/BPEL4WS.jsp>

- IBM <http://www-128.ibm.com/developerworks/library/specification/ws-bpel/>

- MSFT <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnbizspec/html/bpel1-1.asp>

- OASIS WSBPEL TC http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel

[12] Java/J2EE Runtime and Development Environments - examples

- Apache <http://www.apache.org/>

- BEA WebLogic <http://www.bea.com/framework.jsp?CNT=index.htm&FP=/content/products>

- IBM WebSphere, IBM Rational Software Architect, etc
<http://www-306.ibm.com/software/sw-bycategory/>,
<http://www-128.ibm.com/developerworks/platform/index.html>

- JBoss <http://www.jboss.com/>

- Oracle Application Server <http://www.oracle.com/appserver/index.html>

& Oracle Development Tools <http://www.oracle.com/tools/index.html>

Ivan Barać

e-mail: ivan.barac@ericsson.com

Ericsson Nikola Tesla d.d.

Krapinska 45

HR-10002 Zagreb

Hrvatska

Uredništvo je primilo rukopis 8. studenog 2005.