

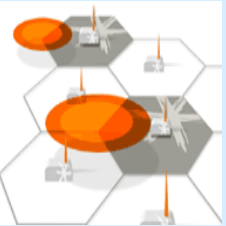

Szolgáltatások és alkalmazások (VITMM131)

Nyílt szolgáltatás hozzáférés, API-k

Vidács Attila
Távközlési és Médiainformatikai Tanszék (TMIT)
I.E.348, T:19-25, vidacs@tmit.bme.hu

Tartalom

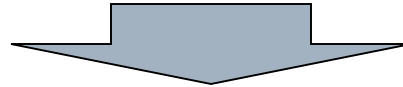
- Áttérés nyílt rendszerekre
- Nyílt szolgáltatás hozzáférés (OSA – Open Service Access)
- Alkalmazás programozói interfész (API – Application Programming Interface (API))
- OSA/Parlay
 - Parlay API-k
 - Parlay X Web szolgáltatások



Áttérés nyílt szolgáltatásokra

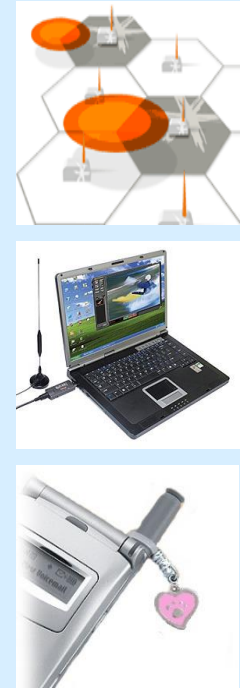
□ Konvergens hálózati trend

- RÉGI: **zárt**, magántulajdonban kezelt rendszerek
 - természeténél fogva rugalmatlan, nem egykönnyen programozható
 - Új szolgáltatások bevezetése gyakran jelentős anyagi és idő- ráfordítást igénylő erőfeszítés.



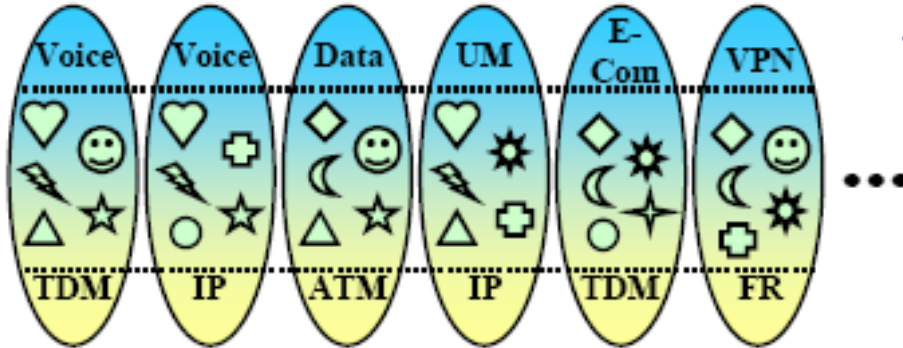
- ÚJ: **nyílt rendszerek** lehetővé teszik bárki számára szolgáltatás nyújtását a felhasználóknak

De milyen „rendszerekről” is van szó?

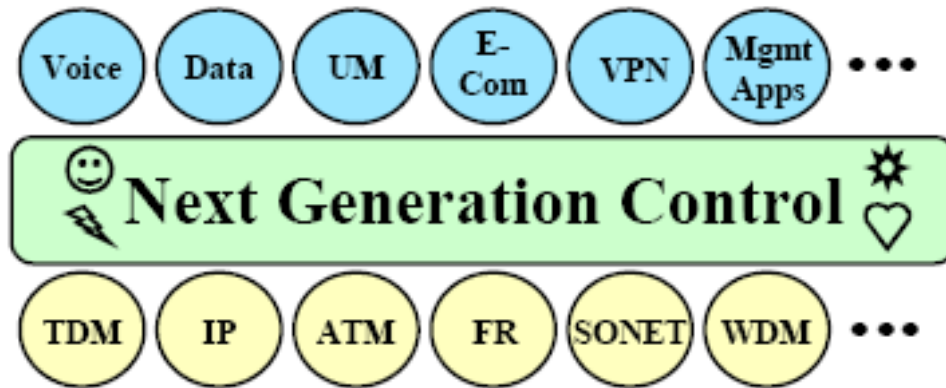


NGN szolgáltatás architektúra

□ NGN megoldás:

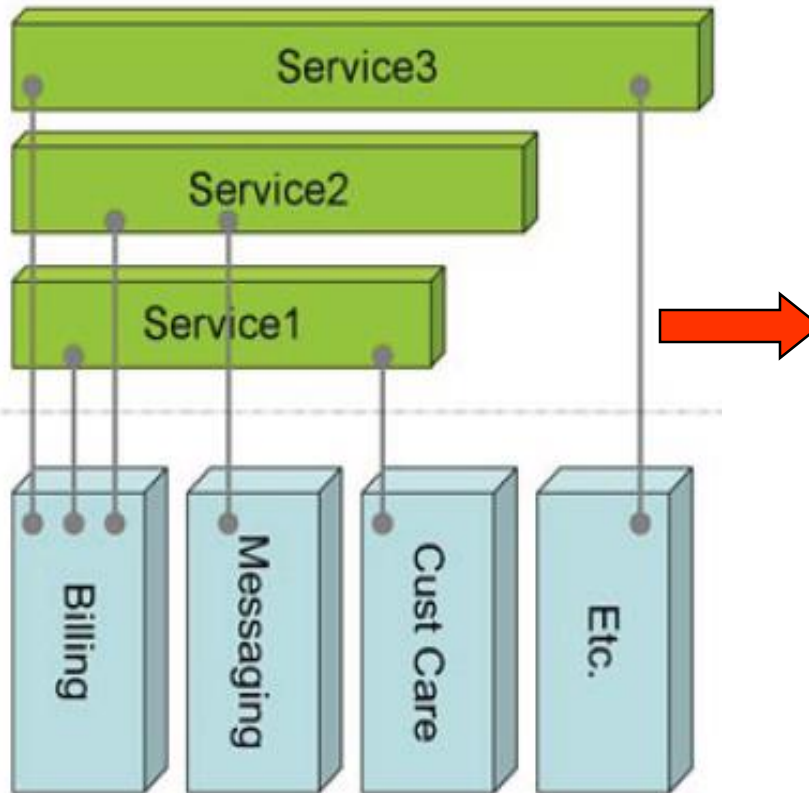


- Jelenlegi megoldások

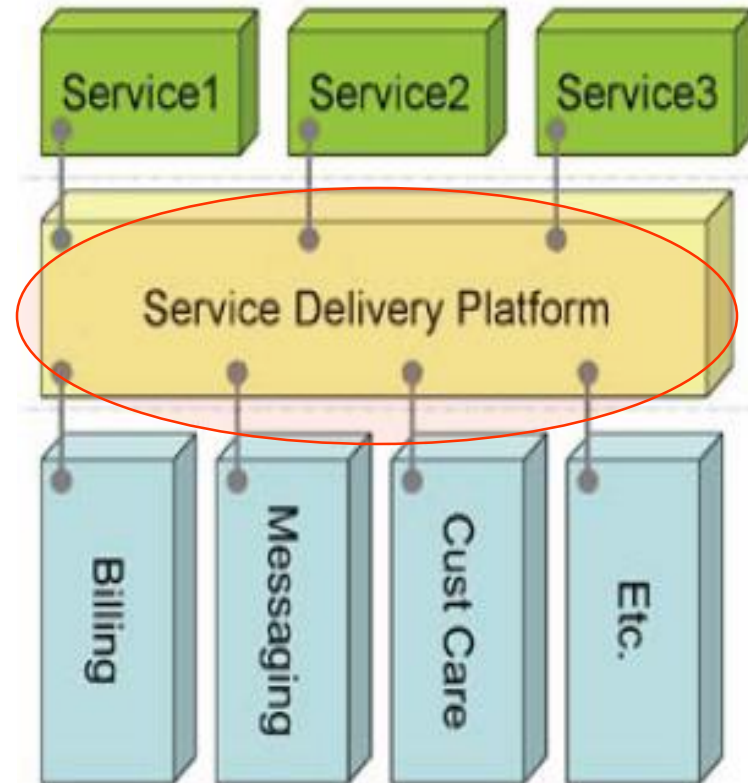


NGN szolgáltatás architektúra (2)


Jelenlegi megoldás:





NGN megoldás:



Áttérés nyílt rendszerekre – „minek?”

- 
- A hálózat megnyitásának **célja**: lehetővé tenni az *alkalmazások* számára a közcélú hálózatok *kommunikációs képességeinek* elérését.

Motivációk:

- 
- 
- A (hálózati) szolgáltatók érdekeltek a forgalom (így a **bevétel!**) növelésében, attraktív szolgáltatások nyújtása által.
 - A **stratégia**: független szoftverfejlesztők bátorítása olyan alkalmazások kifejlesztésére, amelyekkel a (hálózati) kommunikációs szolgáltatások fejleszthetők.
 - Az információs szolgáltatások fejlesztői keresik a hálózati kommunikációs lehetőségeket saját IT alkalmazásaik kiterjesztéséhez.
 - (pl., banki tranzakció értesítők)

Áttérés nyílt rendszerekre – „hogyan?”

□ Mire van szükség?

- Az alkalmazások (azaz felhasználók) *elválasztása* a generikus funkcionalitástól.
- *Biztonságos, megbízható és számlázható* felhasználói kontroll biztosítása a *hálózati funkciók* kezeléséhez, amelyeket különböző hálózati elemek (pl. kapcsolók, adatbázisok) biztosítanak.

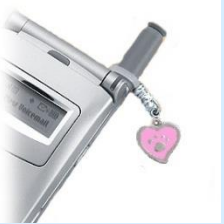
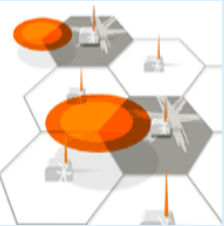
□ Fontos: *Míg az internet természetesen bátorítja és elősegíti (pl. intelligens végberendezések alap átviteli szolgáltatással) a nyílt hálózati felhasználást, addig a távközlési szolgáltatók hálózatainak megnyitásánál figyelembe kell venni a hálózat **szolgáltatási képességeit.***

- Ez megtehető egy **nyílt interfész** biztosításával, azaz az **API** (alkalmazás programozói interfész) koncepció használatával.



Nyílt szolgáltatás hozzáférés (OSA)

- Egy lehetőség a hálózat megnyitására egy *interfész* biztosítása a *generikus szolgáltatási* és az *alkalmazási réteg* között.
- **OSA** (**Open Service Access** – **nyílt szolgáltatás hozzáférés**).
- 3GPP szabvány definíciója szerint: „*az OSA egy koncepció az új szolgáltatások bevezetésének gyártófüggetlen módjára*”.
- A 3GPP szabvány definiál egy **OSA architektúrát** amely lehetővé teszi a távközlési szolgáltató (telco) hálózatán *kívüli* alkalmazás fejlesztők számára, hogy...
 - *kihasználják a hálózati funkcionalitást, és*
 - *információt kapjanak a hálózatból**...szabványos interfészeken keresztül.*
- Egy ún. **nyílt alkalmazás programozói interfész** (API) biztosítja ezt az illesztést.



Az API-k koncepciója

- **API** (**A**pplication **P**rogramming **I**nterface – alkalmazás programozói interfész)
 - *Def: Szolgáltatás képes API-k alatt a hálózat által nyújtott funkciók halmazát értjük, amely lehetővé teszi programozók számára a felhasználóknak szolgáltatást nyújtó programok írását.*

- A hálózati operátorok számára hozzáférhető **nyílt szabványos API-k** konvergens hálózatokban:
 - Lehetővé teszi független szolgáltatás fejlesztők számára a hálózat által támogatott alkalmazások fejlesztését **anélkül, hogy ezzel veszélyeztetnék a hálózatot vagy annak szolgáltatásait.**



Rétegek, protokollok, API-k

- A **távközlő hálózatok** világában a hangsúly a (hálózati) rétegek specifikálásán van.
 - A *protokollok specifikálása és szabványosítása* teszi lehetővé a *hálózatok együttműködését*.
 - Pl. TCP/IP stack: (HTTP + TCP + IP protokollok)
 - Pl. SS7 stack: (ISUP, TUP, MTP-x protokollok)

- A **szoftverek** világában a hangsúly a *rétegek közötti interfészek*en (API – Application Programming Interface) van.
 - *API-k szabványosításával platformfüggetlen alkalmazások* hozhatók létre.
 - Pl. POSIX (Unix és Linux rendszerek)
 - Pl. WIN32 (Microsoft Windows 95/98/2000, stb.)

TCP	Transmission Control Protocol	HTTP	HyperText Transfer/Transport Protocol
SS7	Signalling System Number 7		ISUP ISDN User Part
TUP	Telephony User Part	MTP	Message Transfer Part
POSIX	Portable Operating System Interface		

Rétegek, protokollok, API-k (2)

- Az **API koncepció** használatával a távközlésben lehetőség nyílik *hordozható, technológia- és protokollfüggetlen alkalmazások fejlesztésére*.
 - Pl. konferenciahívás esetén egyes résztvevők PSTN-en, mások GSM-en vagy hálózati terminálokon kapcsolódnak.

- Megjegyzés: Kb. 3 millió professzionális szoftverfejlesztő van a világon, szemben a kevesebb mint 10 ezer specializálódott távközlési szolgáltatás fejlesztővel!



API-k jellemzői

- A **nyílt szolgáltatás hozzáférés API-k** hét **tulajdonsága**:
 - **Nyíltság:**
 - Széles körben elfogadott *szabvány* definiálja.
 - **Biztonság:**
 - A szolgáltatás interfészek hozzáférése csak hitelesített és felhatalmazott fél részére megengedett.
 - **Integritás (sérthetetlenség):**
 - A hálózat funkcionalitása nem veszélyeztethető az alkalmazás esetleges túlzott hívásai következtében.
 - **Rugalmasság:**
 - Az interfész lehetővé teszi változatos alkalmazások támogatását, és az interfész implementációja biztosítja különféle hordozó hálózatok használatát.
 - ...(folyt.)



API-k jellemzői (folyt.)

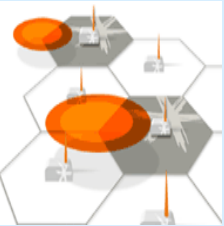
- A **nyílt szolgáltatás hozzáférés API-k** hét **tulajdonsága**:
 - ...
 - **Absztrakció:**
 - A mögöttes hálózat és annak heterogenitása rejtve kell maradjon az alkalmazás programozó előtt. (Azaz a programozótól nem elvárható, hogy megértse az adott hálózati technológiát és jelzési protokolljait.)
 - **Technológia semlegesség:**
 - Az API definíciója nem kötött egyetlen adott nyelvhez vagy implementációs technológiához.
 - **Szolgáltatás felderítés:**
 - Egy alkalmazás képes felderíteni a rendelkezésre álló szolgáltatásokat egy kiszolgáló node-ban.



Az OSA/Parlay

- A bemutatott nyílt API-k három szervezettől eredeztethetők:
 - **A Parlay csoport**
 - a telco-k és gyártók egy konzorciuma, nyílt API-k kifejlesztésének egyedüli céljával;
 - **A 3GPP** (Third-generation partnership project)
 - a harmadik generációs mobil kommunikációs rendszerek minden aspektusával foglalkozik;
 - **ETSI SPAN** (Services and Protocols for Advanced Networks)
 - az ETSI „Szolgáltatások és protokollok fejlett hálózatok számára” „csoportja” OSA és API szabványokat publikál.

- A következőkben tárgyalandó API-k legtöbbször „Parlay/OSA” vagy „OSA/Parlay” néven hivatkoztak.

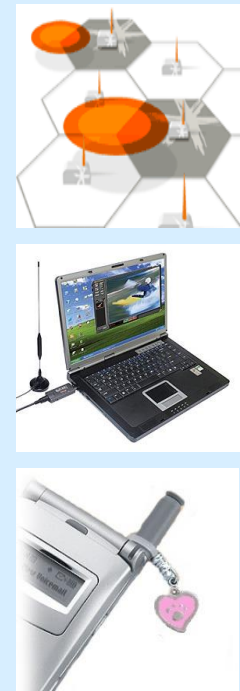


A Parlay csoport

„Welcome to the Parlay Group

*The Parlay Group aims to intimately **link IT applications** with the **capabilities of the telecommunications world** by specifying and promoting **application programming interfaces (APIs)** that are secure, easy to use, rich in functionality, and based on open standards. Parlay integrates telecom network capabilities with IT applications via a secure, measured, and billable interface. Enabled by Parlay's network-independent APIs, applications are generating new revenue streams for network operators, application service providers (ASPs), and independent software vendors (ISVs).”*

www.parlay.org



The Parlay API-k története

- **Cél:** nyílt, technológia független API-k létrehozása a tradicionális hálózati operátor körén *kívüli* alkalmazás fejlesztők számára.
- 1. fázis
 - 1998: A Parlay csoport létrehozása 5 vállalat által: British Telecom, Microsoft, Nortel Technologies, Siemens and Ulticom.
 - 1999. szept: 1.2 verzió – A következő API-k létrehozása: Framework, Call Control, Messaging, és User Interaction
- 2. fázis
 - Hat új tag: AT&T, Cegetel, Cisco, Ericsson, IMB, Lucent Technologies
 - 2000. jan: 2.1 verzió – csomagkapcsolt és vezeték nélküli hálózatok támogatása



The Parlay API-k története

□ 2. fázis (folyt.) :

- 1999: ETSI és 3GPP közös munkába kezdett az OSA (Open System Architecture) kifejlesztésére a Parlay specifikációk felhasználásával.
- ETSI elkezdte a Parlay API-k szabványosítását.

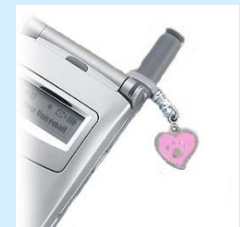
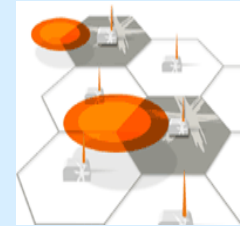
□ 3. fázis:

- 2000 jún: A Csoport kibővült, és hangsúlyt fektetett a Parlay „reklámozására”.
- Együttműködés az ETSI-vel és 3GPP-vel, input a JAIN résztvevőitől → „Joint API Group” megalakítása.
- 2001. júl: *Parlay 3* specifikáció elkészült.

□ 4. fázis:

- 2002. nov: Policy Management, Presence and Availability Management API-k hozzáadása
- Egy új middleware technológia, a **Web szolgáltatások** támogatása

ETSI	European Telecommunications Standards Institute
3GPP	Third Generation Partnership Project
JAIN	Java API for Integrated Networks
OSA	Open Service Access



The Parlay API-k története

□ 4. fázis (folyt.)

■ **Web szolgáltatások:**

- A Parlay API-k WSDL (Web Service Description Language) publikálása, lehetővé téve a Parlay Gateway és az alkalmazás interakcióját HTTP és SOAP (Simple Object Access Protocol) használatával.

- új API garnitúra **Parlay X** néven

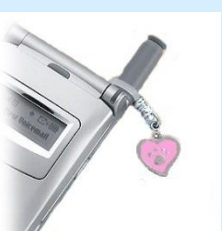
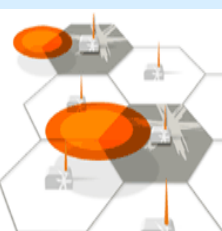
- 2004. jún: *Parlay 4.1* és *Parlay X Web szolgáltatás specifikáció* kiadása

□ fázis... legújabb

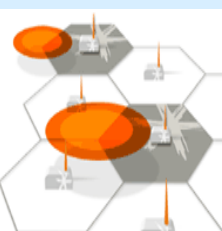
- 2004. szept: *Parlay X 2.0*
- 2005. ápr: *Parlay 5.0*
- 2007. jan: *Parlay 5.1* kiadása
- 2007. nov: Draft *Parlay 6.0 Spec* kiadása (ETSI ES 204 915)
- *2007. szept: Parlay X 3.0*
- 2007-ben befejezte működését.



Parlay csoport – teljes tagok



Parlay csoport – kapcsolattartó tagok



Appium

ARGELA
Technologies

IIA
工業技術研究院
Industrial Technology
Research Institute

CSC
SYSTEMS

KT

logicaCMG
solutions that matter

ERICPOL
TELECOM

ESRI

ETRI

FOKUS

Mobile Cohesion



ftw.

GEMPLUS

HERIT
CORPORATION

hp
invent

OKSIJEN

ORACLE

HUAWEI

infintel

IONA

Kabira

REDWOOD
TECHNOLOGIES

rococo

macnica
networks

Marconi

solomiO

T-Mobile

TDC

TECNOMEN

Net Call

NORTEL
NETWORKS

TEKELEC

telenity

telenor

TeliaSonera

OTE

PERSONETA

TTC

TURKCELL

WANGEL

Ulticom

SBC
SBC Technology
Resources, Inc.

SK Telecom
Most Valuable Company

UTStarcom

vodafone

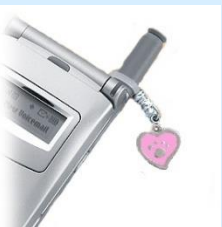
WIPRO
Applying Thought

XENTRONIK

ZTE中兴

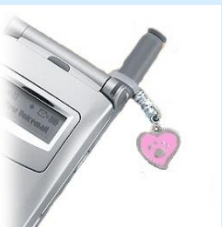
Parlay/OSA architektúra

- A **Parlay/OSA architektúra** két nagy tartományt definiál nyílt API-val a tartományhatáron:
- A **hálózat operátor tartomány**...
 - a *mögöttes hálózat* és a *funkcionalitás*, amely **implementálja a nyílt API-t**;
 - más néven a *telecom szolgáltatói tartomány* (telecom service provider domain).
- A **vállalati operátor tartomány**...
 - egy vagy több *alkalmazást* tartalmaz (kliens alkalmazások) amely a **nyílt API-t használja**.
 - A **vállalati operátor**...
 - egy *független szolgáltató* amely *üzleti kapcsolatban* áll a hálózat operátorral;
 - *feliratkozik* a hálózati operátor által nyújtott *hálózati szintű szolgáltatásokra*.

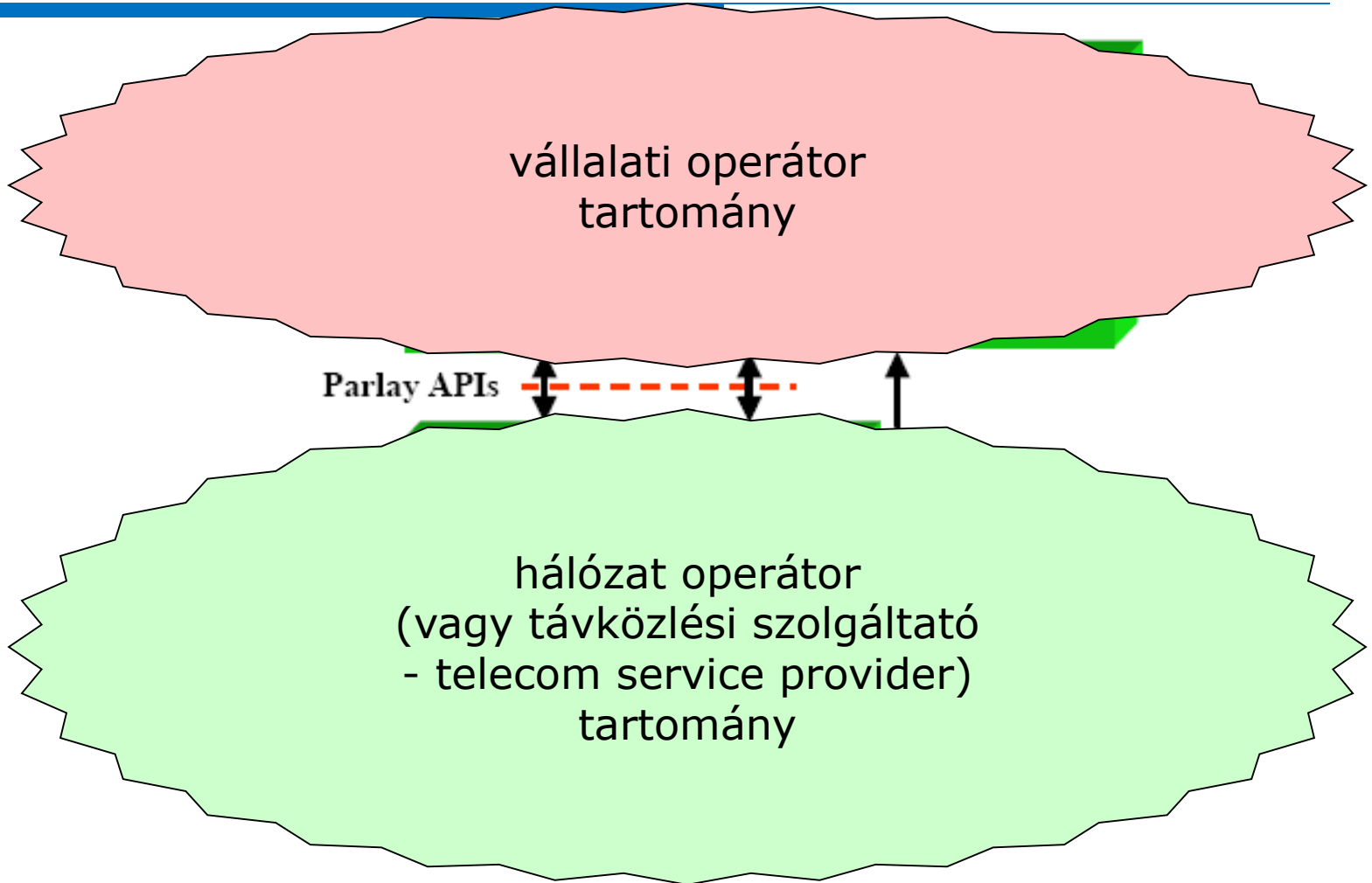


Parlay/OSA architektúra

- A **Parlay/OSA architektúra** két nagy tartományt definiál nyílt API-val a tartományhatáron:
- A **hálózat operátor tartomány**...
 - a *mögöttes hálózat* és a *funkcionalitás*, amely **implementálja a nyílt API-t**;
 - más néven a *telecom szolgáltatói tartomány* (telecom service provider domain).
- A **vállalati operátor tartomány**...
 - egy vagy több *alkalmazást* tartalmaz (kliens alkalmazások) amely a **nyílt API-t használja**.
 - A **vállalati operátor**...
 - egy *független szolgáltató* amely *üzleti kapcsolatban* áll a hálózat operátorral.;
 - *feliratkozik* a hálózati operátor által nyújtott *hálózati szintű szolgáltatásokra*.



Parlay/OSA architektúra



forrás: parlay.org

Parlay/OSA Framework

- A hálózati operátorok és szolgáltatók részéről kulcskövetelmény, hogy a hálózat megnyitásával *a hálózati infrastruktúra ne legyen kiszolgáltatva jogosulatlan hozzáféréseknek* vagy egyéb fenyegetéseknek.
- Ezt a funkciót a **Framework API** látja el, amely egy *Parlay/OSA Gateway-en* (átjárón) implementált.



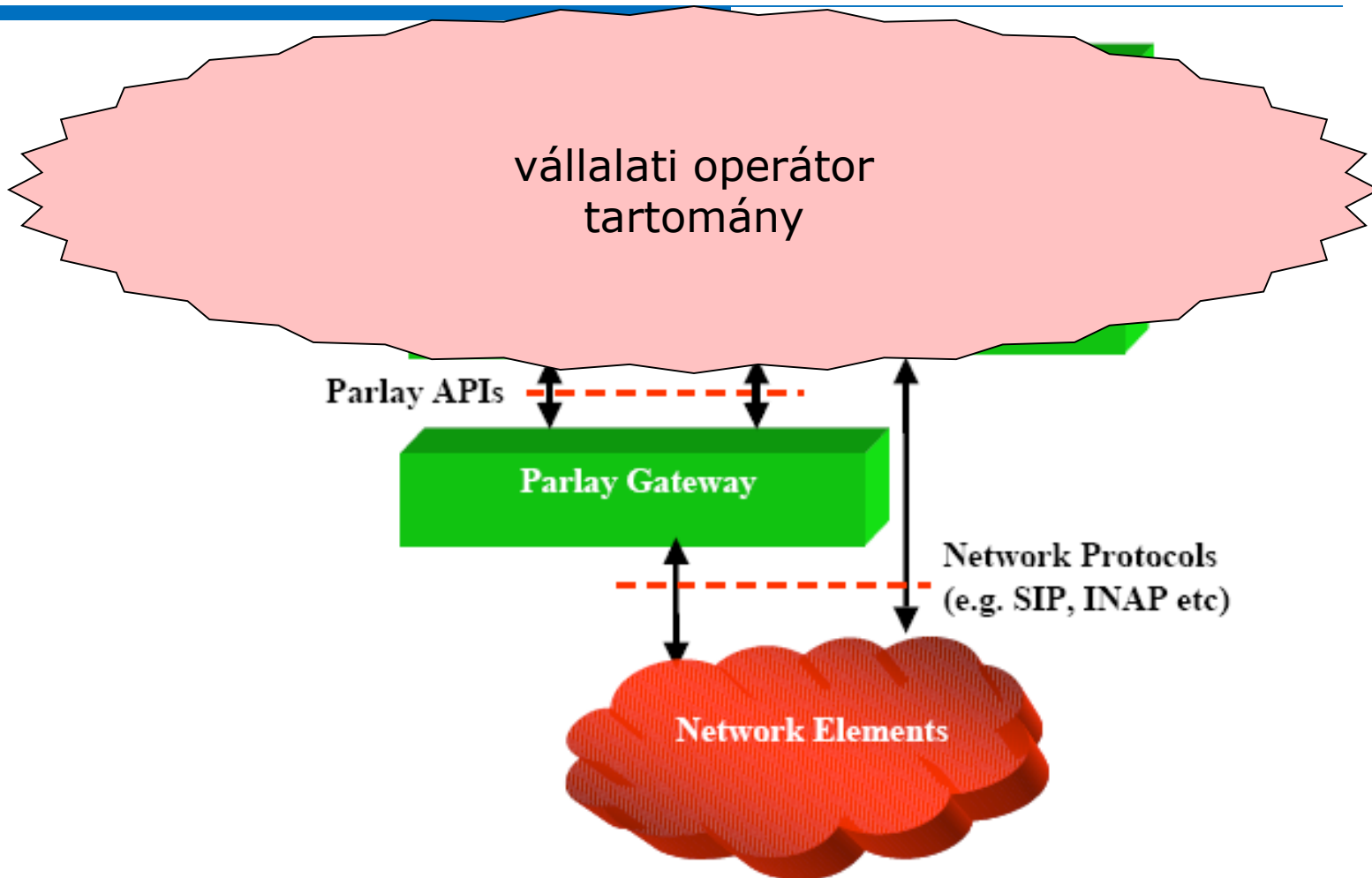
Parlay/OSA Gateway

- A Parlay/OSA **Gateway** egy logikai entitás, amely a *Framework API*-t valósítja meg. Fizikai megvalósítása így szabadon választható, pl:
 - Önálló, különálló gateway, amely közvetlenül a hálózati elemekhez kapcsolódik.
 - Pl. „Jambala” Network resource Gateway (Ericsson)
 - Pl. „MiLife” Parlay Gateway (Lucent)
 - Pl. Causeway Parlay Gateway (AePONA)
 - Létező intelligens hálózati (IN) platformba (pl. egy SCP-n belül) integrált gateway.
 - Pl. Application Mediation Gateway (Marconi) a System-X SCP-ben
 - Pl. Open Service Gateway (Telcordia) mint az ISCP platform része

SCP

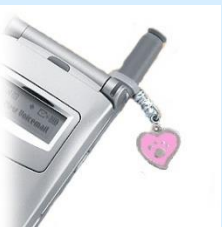
Service Control Point

Parlay/OSA architektúra

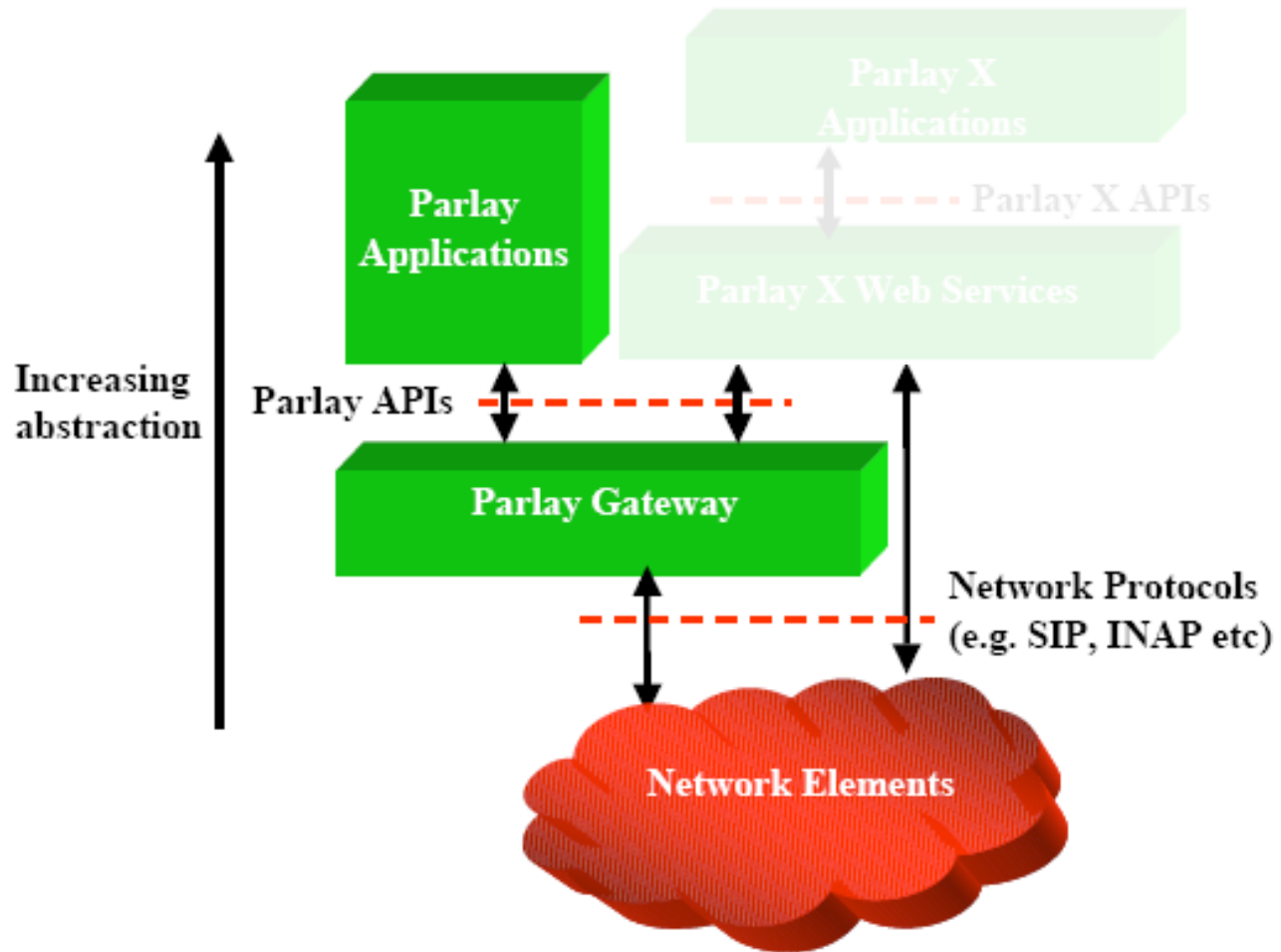
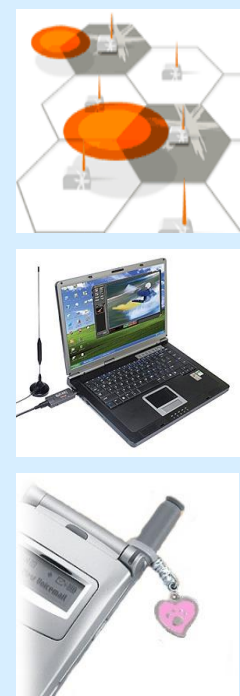


Parlay/OSA Application Server

- Az *alkalmazás* egy **alkalmazás kiszolgálón** (**Application Server**) fut, és TCP/IP hálózaton kapcsolódik a *gateway*-hez.
- Vállalati alkalmazás kiszolgálók elhelyezési lehetőségek:
 - A kiszolgáló egy *vállalati adatközpontban van*, valamilyen üzleti alkalmazást támogat, és a távközlési szolgáltató hálózatát használja a Parlay API-n keresztül.
 - Egy vállalaton kívüli *harmadik fél* (pl. alkalmazás szolgáltató) ad otthont a kiszolgálónak a vállalat megbízásából.
- Sok szolgáltató telepít alkalmazás kiszolgálókat a saját hálózatán belül.
 - Ebben az esetben u.azokat a Parlay/OSA API-kat használja, mint bármely harmadik fél.



Parlay/OSA architektúra



SIP
INAP

Session Initiation Protocol
Intelligent Network Application Protocol

forrás: parlay.org

Parlay API-k

- A **Parlay API-k** két fő részre oszthatók:

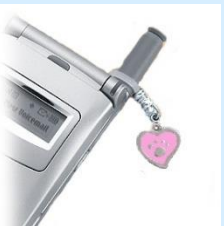
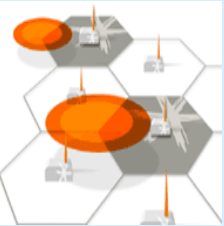
- „Framework” (**keretrendszer**)
- „Services” (**szolgáltatások**)

- A **Framework** rész...

- Biztosítja a *lehetőséget* a hálózati operátor és az alkalmazást nyújtó fél közötti *egyeztetésre*.
- Biztosítja a *kapcsolatfelvételi pontot* az alkalmazás számára a hálózat által nyújtott szolgáltatások felderítéséhez.
- Minden szükséges *biztonsági óvintézkedést* figyelembe vesz.

- A **Services** rész...

- Nagyszámú *interfészt* biztosít a hálózat képességeinek és lehetőségeinek hozzáféréséhez és kihasználásához.
(Pl. mobilitás menedzsment, hívásvezérlés, jelenlét és elérhetőség, ...)



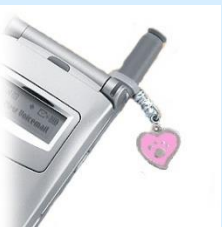
Parlay API-k (2)

- Parlay v4.0, 3GPP OSA rel.5, ETSI ES 202 915 API-k:
 - General
 - Common Data
 - **Framework**
 - Call Control
 - User Interaction
 - Mobility
 - Terminal capabilities
 - Presence and Availability Management
 - Policy management
 - Generic Messaging
 - Content Based Charging
 - Data Session Control
 - Account Management
 - Connectivity Management



Parlay/OSA Terminology: SCSs and SCFs

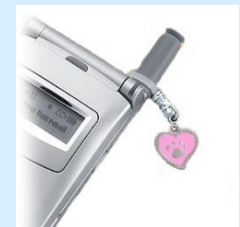
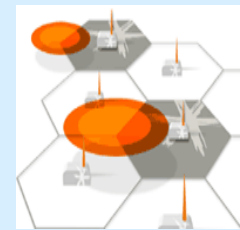
- The Parlay/OSA Gateway consists of several **Service Capability Servers (SCS)**: functional entities that provide Parlay/OSA interfaces towards applications.
- Each SCS is seen by applications as one or more **Service Capability Features (SCF)**: abstractions of the functionality offered by the network, accessible via the Parlay/OSA API. Sometimes they are also called **services**
- The Parlay/OSA SCFs are specified in terms of interface classes and their methods



Parlay API-k

□ **Framework**

- az alap mechanizmusokat biztosítja az alkalmazások számára a hálózat által felajánlott szolgáltatások igénybevételéhez.
- Mielőtt az alkalmazás használhatná az API-t, *hitelesítésre* van szükség az *alkalmazás* és a *Framework* között.
- **Framework interfészek:**
 - Megbízhatóság és biztonság menedzsment – Trust and security management
 - Regisztráció – Registration
 - Szolgáltatás felderítés – Service discovery
 - Szolgáltatás életrajz menedzser – Service lifecycle manager
 - Integritás menedzser – Integrity management
 - Esemény értesítés – Event notification
 - Szolgáltatási megállapodás menedzsment – Service agreement management
 - Szolgáltatás előfizetés – Service subscription



Parlay API-k (3)



□ **Call Control** (hívásvezérlés)

- Lehetővé teszi az alkalmazásoknak *hívások felépítését* a hálózatban, alapvető hívásvezérlési és többrésztvevős hívásvezérlési funkciókat is biztosítva.



□ **User Interaction** (felhasználói interakció)

- Definiálja, hogyan juthatnak az alkalmazások információhoz a *végfelhasználótól*, közleményeket kezel, rövid üzeneteket küld, stb.



□ **Mobility** (mobilitás)

- Lehetővé teszi az alkalmazások számára, hogy meghatározzák egy terminál (telefon) *elhelyezkedését*, és lehetővé teszi számukra, hogy értesítést kérjenek amikor a terminál helyet változtat.

□ **Terminal** (végberendezés)

- Lehetővé teszi a felhasználói végberendezés (telefon) *képességeinek felderítését*.
-

Parlay API-k (4)

□ **Availability Management and Presence** (**elérhetőség és jelenlét**)

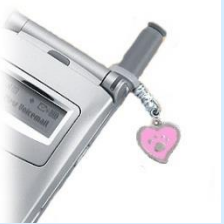
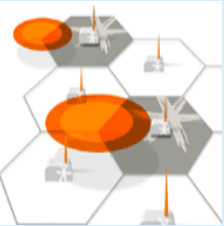
- Lehetővé teszi az alkalmazások számára, hogy információt kapjanak (ill. információval szolgáljanak) a felhasználók *elérhetőségéről* és *jelenlétéről*. (pl. „Otthon vagyok.”, „Az irodámban ülök.”)

□ **Policy Management** (**felügyelet**)

- *Felügyeleti szabályok* létrehozásához, vagy *felügyeleti események* regisztrálásához

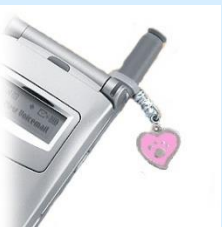
□ **Messaging** (**üzenetküldés**)

- Hozzáférést engedélyez postafiókokhoz, *üzeneteket küld és fogad*, alkalmazások interakcióját kezeli az *üzenetkezelő rendszerekkel* (pl, hang, fax, e-mail, ...).



Parlay API-k (5)

- **Content Based Charging** (tartalomfüggő számlázás)
 - Vezérli az alkalmazások kéréseit a szolgáltatás *kifizetéséről*.
- **Data Session Control** (adathívás vezérlés)
 - Menedzseli az alkalmazás terminálról indított *adatforgalmát*. (Tipikusan GPRS és 2.5G alkalmazások esetén.)
- **Account Manager** (könyvelés)
 - Lehetőséget ad az alkalmazásoknak hozzáférési információk és múltbéli *tranzakciók listázására*.
- **Connectivity Management** (kapcsolat)
 - Lehetővé teszi az alkalmazások számára a végpontok közötti *szolgáltatminőség (QoS)* befolyásolását ill. vezérlését.



A Parlay használata

Parlay
Application
Server

Applications

Services
Admin

3
Framework

4
Service

Parlay Gateway

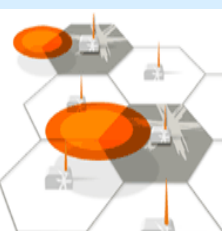
1

2

5

Telecom Voice
Infrastructure

Telecom Data
Infrastructure



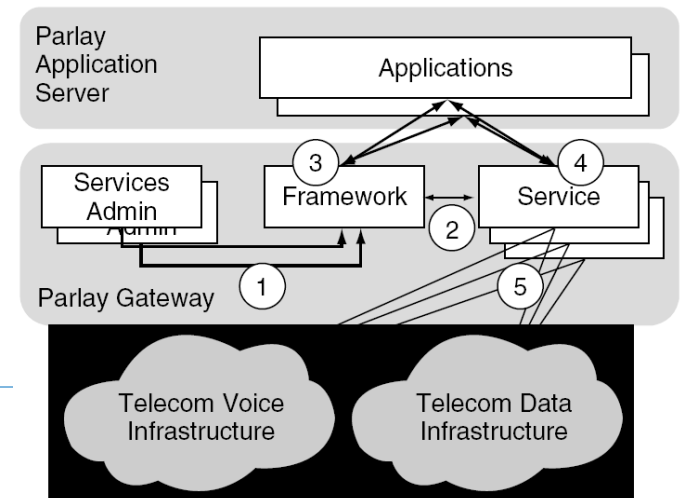
A Parlay használata

□ 1. lépés

- A „harmadik fél” szolgáltató **kifejleszthet** egy Parlay szolgáltatást, majd **regisztrálhatja** azt a Parlay *Framework*-ben.
- A Parlay *Framework* **nyújtja** a felderítés, azonosítás, hitelesítés és szolgáltatás megállapodás funkciókat a Parlay szolgáltatás számára.

□ 2. lépés

- A Parlay szolgáltatás **kezdeményez** egy felhasználói viszonyt a *Framework*-kel.
- A felhasználói viszony alatt a szolgáltatás **elérhető** a Parlay alkalmazás számára.



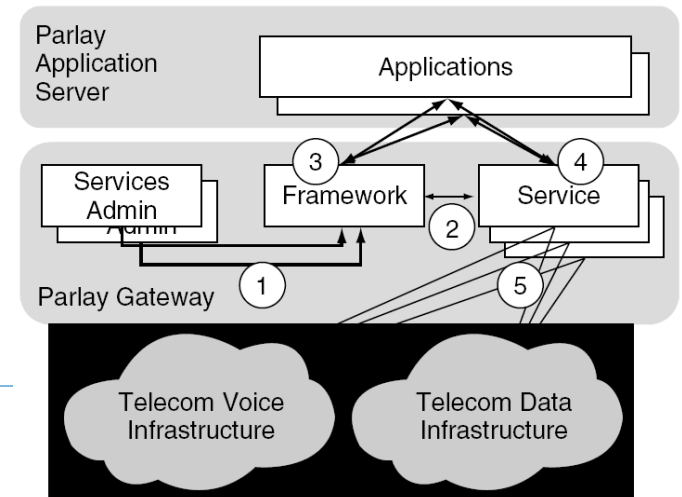
A Parlay használata

□ 3. lépés

- A Parlay alkalmazás **hitelesítést** és **szolgáltatási megállapodást kezdeményez**. (A megállapodásban az alkalmazás megjelöli a vállalt feltételeit és az elvárt biztosítékokat.)
- Az operátor **elfogadja** vagy **elveti** a szolgáltatási megállapodást.
- Ha elfogadja a szolgáltatási megállapodást, egy **referenciát ad vissza a szolgáltatásra**.

□ 4. lépés

- Az alkalmazás **hozzáférhet a szolgáltatáshoz**.



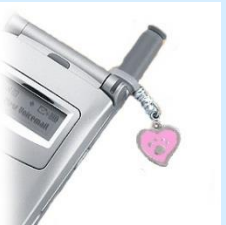
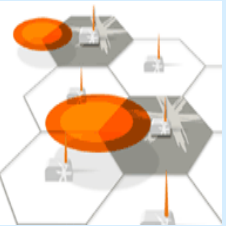
Parlay/OSA szolgáltatási példák

□ 1. példa: Elérhetőség vezeték nélküli hálózatban.

- Pl. „Alíz egy megbeszélésen vesz részt, nem tud telefonhívást fogadni, de tud szöveges üzeneteket olvasni.”
- Parlay/OSA szolgáltatások: „Presence and Availability management”, „Location”, „Call Control”, „User Interction”

□ 2. példa: „Push” szolgáltatás.

- Egy mobiltelefonra szabott weboldal küldése a hívó félnek a hívás elején vagy utána.
- Pl. Ha valaki pizzát rendel, tájékoztatást kap az épp aktuális ajánlatokról.
- Pl. Egy üzleti telefonbeszélgetés után a hívott elküldi elektronikus névjegykártyáját a hívónak.



Parlay/OSA szolgáltatási példák (2)

□ 3. példa: **Számlázás harmadik félnek.**

- Pl. (holland KPN szolgáltatása): „Kapcsoljuk össze A-t B-vel és számlázzunk érte C-nek”.
- Pl. Kormányzati megrendelésre idős emberek árurendelésének telefondíját átvállalja a kormányzat.

□ 4. példa: **Egy felhasználó összeállítja barátainak listáját, majd értesítést kér arról, amikor ezek az emberek a környezetében tartózkodnak. Ha a listából egy bizonyos kör jelen van, egy többrésztvevős hívást szeretne kezdeményezni, vagy üzenetet küldeni mindnekinek.**

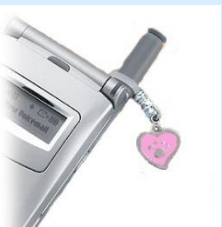
(részletesen ld. köv. fólia...)



Parlay/OSA szolgáltatási példák (3)

□ 4. példa – részletesen:

- Az alkalmazás kapcsolódik a Framework-höz, hitelesíti magát, majd hozzáférést kér a felhasználó státuszához, elhelyezkedéséhez, hívásvezérléshez és felhasználói interakció API-hoz.
- Ha az alkalmazás számára engedélyezett a kért API-k hozzáférése, a Framework utasítja az SCS-eket egy objektum példányának létrehozására az alkalmazás számára, majd elküldi ezek referenciáját az alkalmazásnak.
- A felhasználó ezek után elérheti az alkalmazást (pl. interneten, WAP-on), összeállíthatja a barátok listáját és meghatározhatja az értesítés feltételeit.
- A lista alapján az alkalmazás értesíti a felhasználó státusz/elhelyezkedés SCS-eket és értesítést kér ha a listán szereplők felbukkannak.



Parlay/OSA szolgáltatási példák (4)

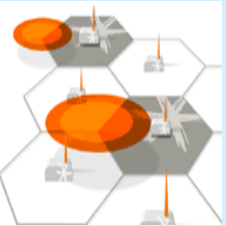
□ 4. példa – részletesen (folyt.):

- Az SCS „trigger”-eket állít be a hálózatban minden személyre.
- Minden alkalommal amikor egy keresett személy felbukkan, a felhasználó státusz/elhelyezkedés SCS értesítést kap.
- A státusz/elhelyezkedés SCS értesíti az alkalmazást.
- Ha az alkalmazás úgy dönt, hogy kellő számú barát jelen van, értesíti a felhasználót (pl. megnyit egy ablakot, vagy a *hívásvezérlés SCS* segítségével felhívja, vagy a *felhasználói interakció SCS*-t használva sms-t küld neki).
- A felhasználó elhatározza, hogy konferenciahívást hoz létre a barátainak.
- Az alkalmazás ellenőrizheti, hogy van-e elég kreditje a felhasználónak a számlázó SCS megkérdezésével.
- Az alkalmazás létrehozza a konferenciahívást a *hívásvezérlés SCS* segítségével.

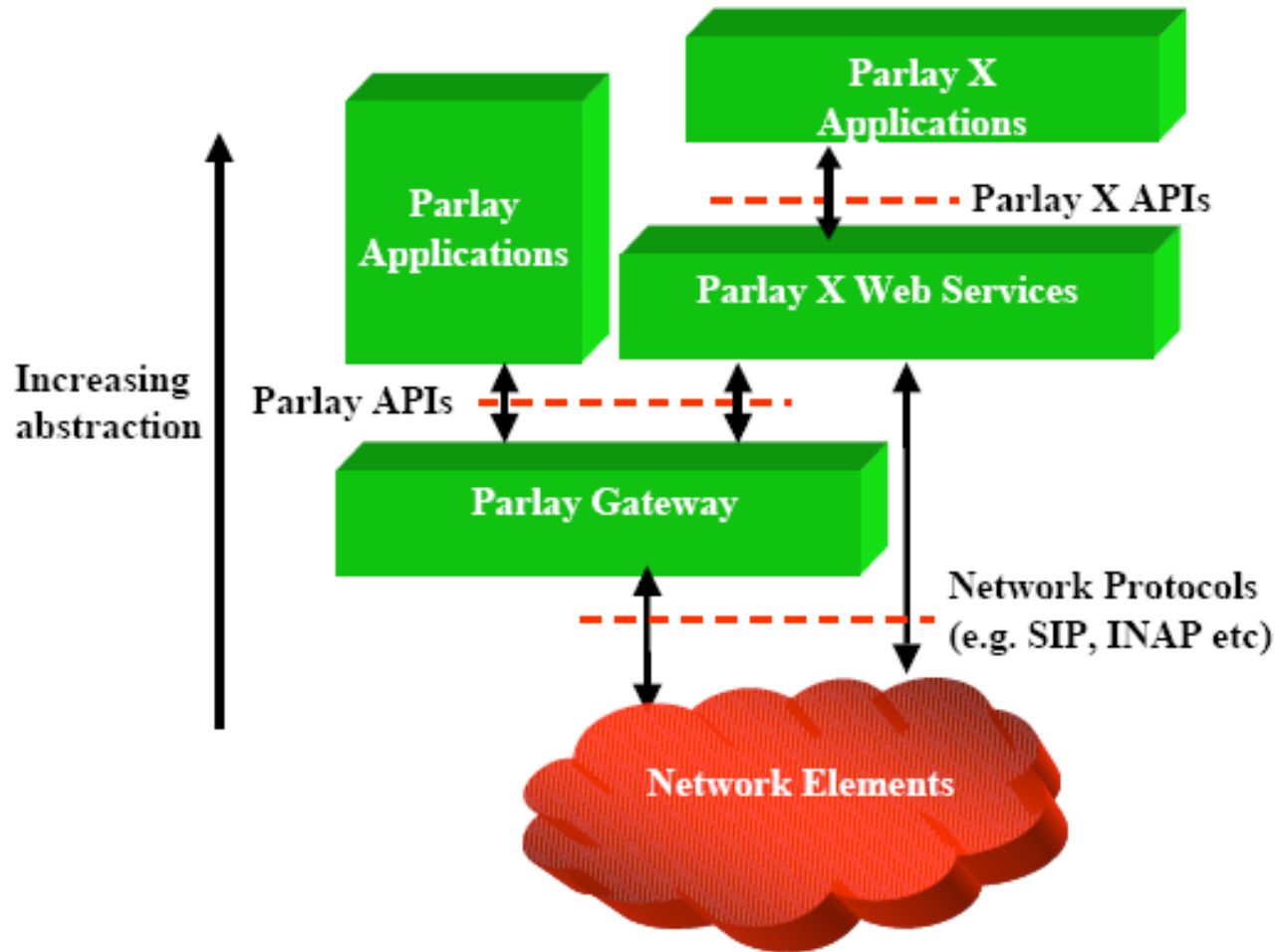
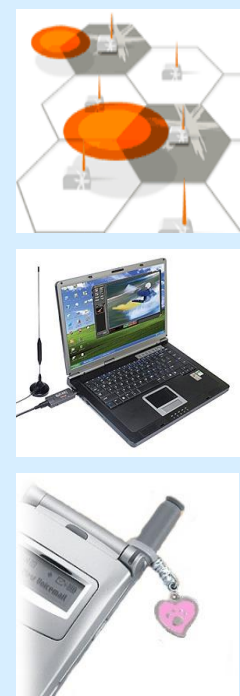


Tartalom

- Áttérés nyílt rendszerekre
- Nyílt szolgáltatás hozzáférés (OSA – Open Service Access)
- Alkalmazás programozói interfész (API – Application Programming Interface (API))
- OSA/Parlay
 - Parlay API-k
 - **Parlay X Web szolgáltatások**



Parlay/OSA architektúra



SIP
INAP

Session Initiation Protocol
Intelligent Network Application Protocol

forrás: parlay.org

Parlay X – Miért?

□ Az OSA/Parlay API...

☺ absztrakt módon fogalmazza meg a hálózat funkcionalitását, így a programozónak **nem szükséges megértenie a mögöttes hálózati protokollokat,**

☹ de az **SCF által meghatározott szinten kell dolgoznia.**

□ Pl.: Az adattípusok széles skáláját kell megértenie és alkalmaznia az alkalmazás programozónak.

□ Más szavakkal: az OSA/Parlay API...




☺ **nagyon gazdag metódusokban** a hálózati részleteket elrejtő absztrakciós szinten, de

☹ a hívások, üzenetek és adatbázis műveletek **részletes kontrollját követeli meg.**

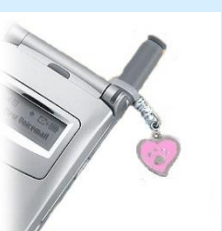
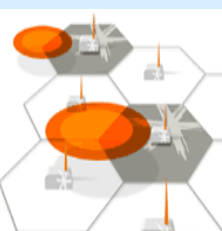
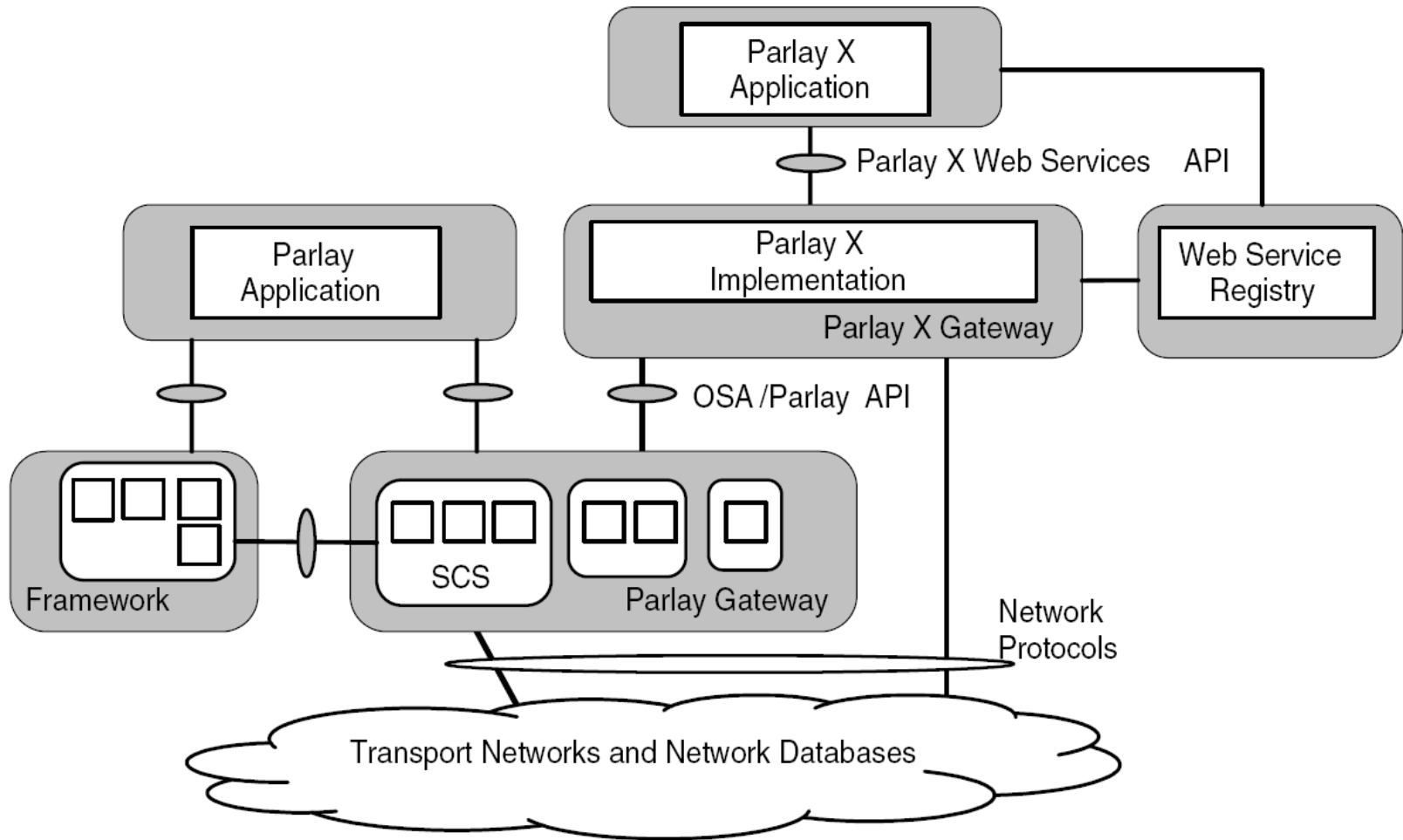
□ *Ennek következtében erős érvek szólnak egy **második OSA interfész** mellett, amely még **magasabb szintű absztrakciós szintet biztosít**, elrejtve a hívás, üzenetküldés és adatbázis műveletek apró részleteit is.*



Parlay X Web szolgáltatások

- 
- 
- 
- A Parlay Csoport definiálta a **Parlay X Web szolgáltatások**-okat mint egy egyszerűsített, magasabb absztrakciós szintű (így, következésképpen, behatárolt) módját a hálózati funkciók eléréséhez.
 - **Parlay X** nem más, mint **Web szolgáltatások specifikálása nyílt szolgáltatás hozzáféréshez (OSA)**.
 - A (WSDL-ben definiált) Parlay X interfészek *ETSI szabványként* publikáltak.
 - Megjegyzés: Érvek a Parlay X mellett
 - Kb. **10,000 szolgáltatás fejlesztő** van világszerte, akik képesek **hálózati protokoll** (e.g., ISUP, MAP, INAP) **szinten** programozni;
 - Kb. **250,000**, CORBA környezetben C-ben vagy Java-ban jártas **programozó** képes a szükséges szinten programozni **OSA/Parlay API-k használatával**;
 - **Programozók milliói** képesek **Web szolgáltatási környezetben** dolgozni.

Parlay X architektúra



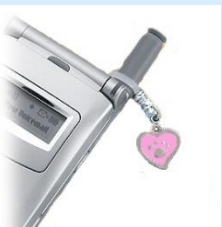
Parlay X architektúra (folyt.)

- A *Framework, Gateway* és *Parlay alkalmazások* tartományok változatlanok.
- A *Parlay X Gateway* a hozzáadott elem
 - Lényegében egy *Parlay alkalmazás* amely Web szolgáltatás interfészeket nyújt az alkalmazások széles osztályának, az ún. *Parlay X alkalmazásoknak*.
 - Megjegyzés: Néhány Web szolgáltatás implementáció nem feltétlen használja a teljes Parlay Gateway-t, és használhat direktben protollokat!
 - Pl. e-mailek küldése
- Megjegyzés: A Parlay X szabványok NEM definiálnak egy framework-szerű biztonsági mechanizmust!
 - Helyette Web szolgáltatás biztonsági előírásokat kell követni.



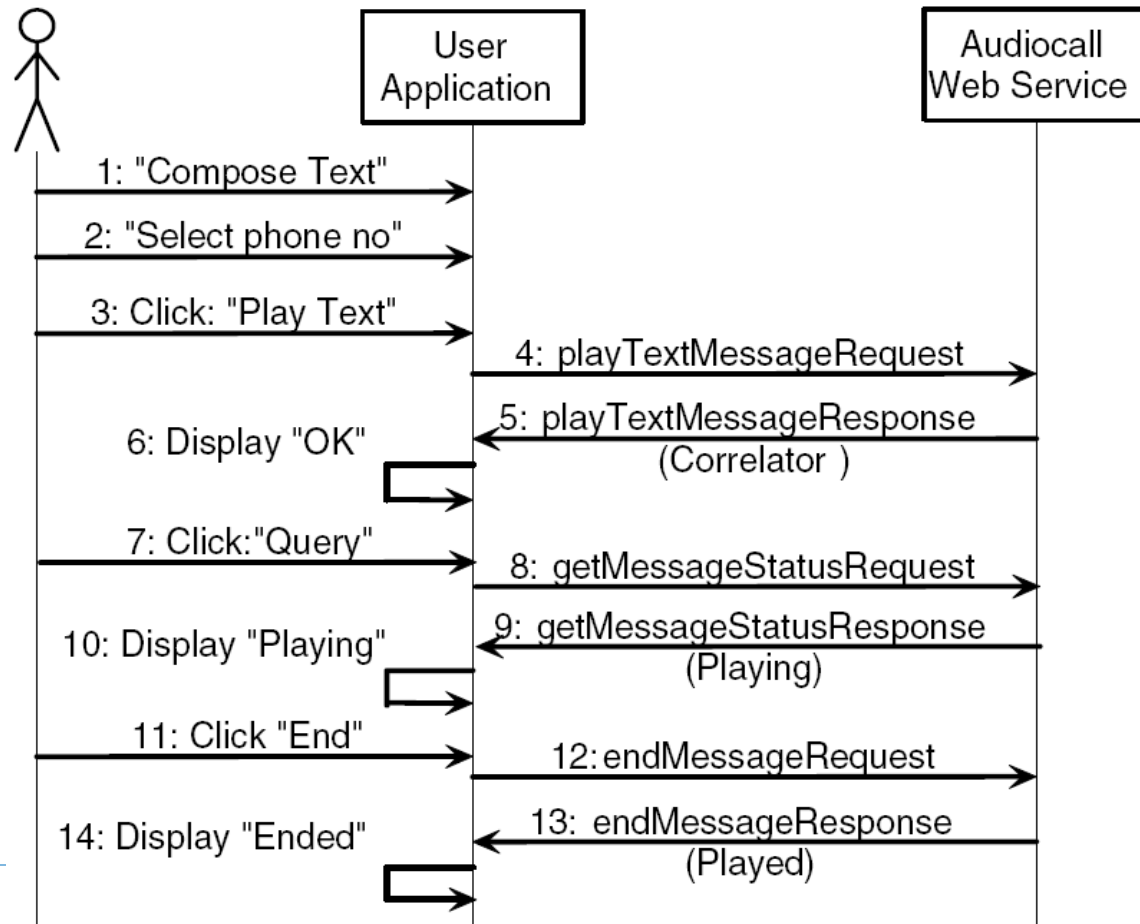
Parlay X Web szolgáltatások

- **Parlay X Web szolgáltatások** (2005-ös szabványokban definiáltak) két csoportba oszthatók:
 - **Hívás és konferenciával kapcsolatos Web szolgáltatások:**
 - két résztvevős hívások létrehozása, hívás státuszának lekérdezése, hívás felbontása;
 - multimédia konferencia vezérlése;
 - rögzített hanghívás létrehozása, azaz felhasználó felhívása és tartalom lejátszása;
 - rövid üzenet küldése;
 - tartalomért fizetés;
 - terminál elhelyezkedésének és státuszának lekérdezése és opcionálisan értesítése.
 - **Szolgáltatás menedzsment:**
 - számlák menedzsmentje (számlál/egyenleg feltöltése);
 - címlisták menedzsmentje: „fehér” és „fekete” listák, átirányított címek;
 - hívások kezelésének specifikálása: feltételek, bejelentések;
 - felhasználói jelenlét információk kinyerése és regisztrálása: aktivitás, hely, környezet, kommunikációs mód.



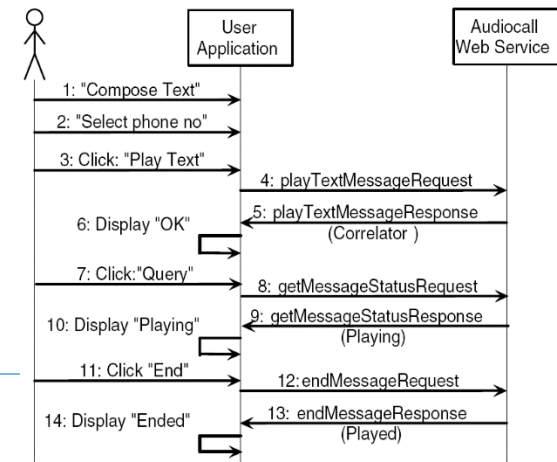
Parlay X Web szolgáltatás példa

- Példa: Egy (emberi) felhasználó egy szöveges üzenetet szeretne lejátszani egy másik felhasználó telefonján „text-to-voice” konverter használatával.



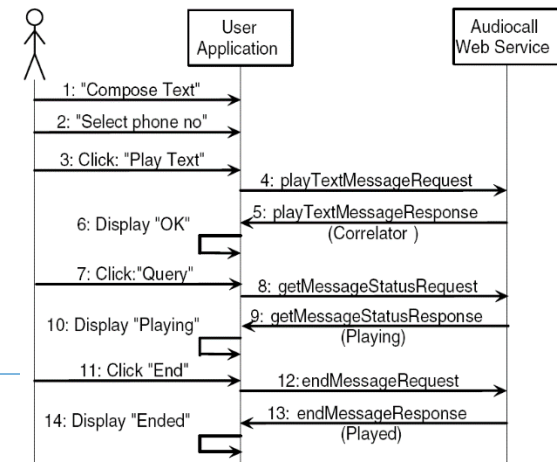
Parlay X Web szolgáltatás példa (folyt.)

- 1, 2. A felhasználó létrehoz egy szöveges üzenetet grafikus felhasználói felületen (GUI), és kiválasztja a felhívandó fél telefonszámát.
3. A felhasználó rákattint egy gombra az üzenet lejátszásához.
4. Az alkalmazás meghívja a **Audiocall Web szolgáltatást** a *PlayTextMessageRequest* üzenet használatával. (A használandó nyelv és a számlázási mód is az üzenetben meg van adva.)
5. A *PlayTextMessageResponse* üzenet azonnal visszaérkezik. Egy azonosító Correlator mint visszakapott érték használható az adott hívás későbbi hivatkozására. A hívás feldolgozódik a Web szolgáltatás implementációja jóvoltából.
6. Egy megerősítő üzenet jelenik meg a GUI-n.



Parlay X Web szolgáltatás példa (folyt.)

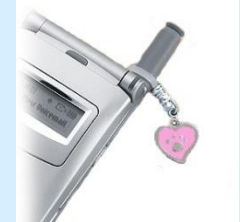
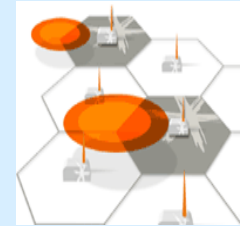
7. Bizonyos idő elteltével a felhasználó szeretné ellenőrizni a hívás folyamatának állapotát egy gombra kattintva a GUI-n.
8. Az alkalmazás meghívja a *GetMessageStatusRequest* üzenetet, a correlator segítségével azonosítva a hívást.
- 9, 10. A hívás visszaadott státusza: „lejátszás alatt”. Egyéb visszatérési érték lehet: “pending” és “error”. A lejátszás státusza megjelenik a GUI-n.
11. A felhasználó eldöntheti, hogy mikor állítja le a hívást.
- 12–14. Az alkalmazás meghívja az *EndMessage* műveletet. Az implementáció interakcióba kezd a gateway-jel a hanghívás törlésére. A hívás lezárása megjelenik a GUI-n.



Parlay/OSA absztrakció – ismétlés

- Egy illusztratív metafóra:

hálózati protokollokkal dolgozni	↔	assembly-ben programozni
OSA/Parlay API használata	↔	magas szintű programnyelvek
Parlay X használata	↔	szkriptek



Parlay/OSA absztrakció – ismétlés (folyt.)

Signalling Protocol	OSA/Parlay API	Parlay X API
Messages, Methods and Operations Call/session signalling messages	Rich, fine grain requests, responses, notifications	Rich functionality, coarse grain operations, no notifications
Calls, procedures and states Follows protocol state machine and procedures	Numerous elementary use cases, asynchronous, stateful, events	Synchronous, simple, stateless, status by polling
Data structures Complex, specific to protocol	Complex, typed with inheritance	Simple XML schema
Invocation of value added services Relies on triggering on message data fields	First or third party invocation	Third party invocation
Knowledge required Protocol details, call processing, media descriptions, ...	Concepts of call model, conference, message, mailbox, ...	Service: call made, message sent, preferences set, ...



Szolgáltatás telepítés scenáriók

