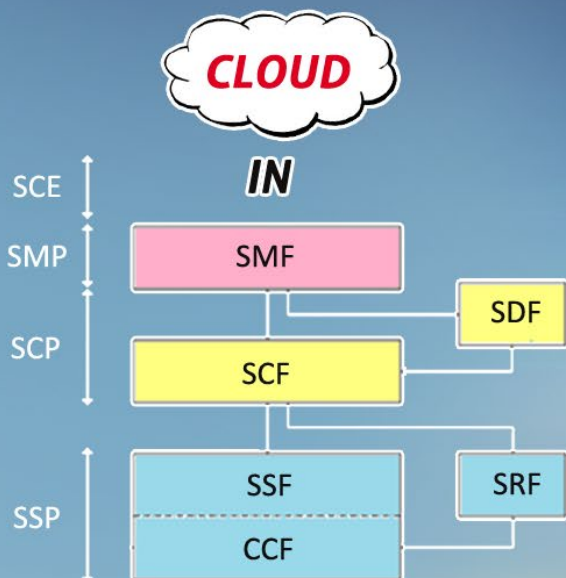


REȚELE INTELIGENTE DE TELECOMUNICAȚII



Dan Nicolae ROBU
Cosmin COSTACHE
- editori -

D.N.Robu

C.Costache

- editori -

REȚELE INTELIGENTE DE TELECOMUNICAȚII

Editura TRANSILVANIA

ISBN: 978-606-19-0820-2

Brașov - 2016

Cuprins

Pag.

1	Cuvânt înainte	- <i>D.Robu, C.Costache</i>
4	Contextul Rețelelor Inteligente de Telecomunicații	- <i>V.Cazacu, D.Robu, T.Bălan</i>
59	Modelul de Procesare a Apelurilor	- <i>S.Bâlbâe, F.Sandu, T.Bălan</i>
85	Modelul de Controler al Rețelei Inteligente	- <i>I.Rîpea, F.Sandu</i>
99	Servicii Tipice în Rețelele Inteligente - Implementare	- <i>V.Cazacu, D.Robu</i>
127	Software pentru Crearea de Servicii	- <i>C.Costache, V.Cazacu, G.Steala</i>
135	JAIN - Java for Intelligent Networks	- <i>S.Dovăncescu</i>
182	Mediul de Creare a Serviciilor pentru Rețele Inteligente de Telecomunicații	- <i>R.David, C.Costache, F.Sandu</i>
249	Implementarea apelurilor cu extensia IBM Rational pentru SIP	- <i>F.Sandu</i>
266	Comunicații Cloud - Abordarea orientată pe servicii	- <i>C.Costache, G.Steala, D.Robu</i>
314	WebRTC - Comunicații prin web în timp real	- <i>G.Steala, Ș.Șurariu, T.Bălan</i>
364	Managementul Business al Comunicațiilor	- <i>A.Balica, F.Sandu</i>
435	Bibliografie	
447	Lista abrevierilor	

Software pentru crearea de servicii

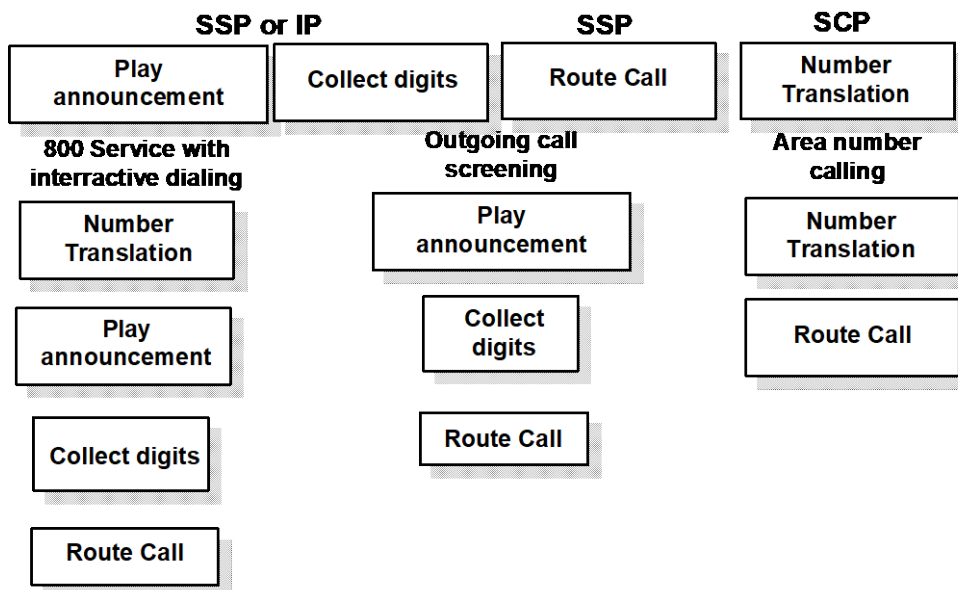
C.Costache, V.Cazacu, G.Stealea

Abordarea cu blocuri constructive independente de servicii (SIB)

Denumirea de blocuri de bază independente de serviciu vine de la “Service Independent building Blocks, sau mai pe scurt, SIBs. SIB-urile sunt reprezentarea abstractă a posibilităților IN și sunt folosite pentru a separa serviciile de tehnologiile folosite la implementarea lor.

Controlul normal al unui apel este reprezentat sub forma unui SIB special și anume blocul procesului apelului de bază (BCP = Basic Call Process), care are în plus capacitatea să decidă când anume este nevoie de funcționalitatea IN-ului.

Serviciile sunt realizate prin înlănțuirea SIB-urilor care preiau controlul de la SIB-ul BCP atunci când funcționalitatea IN este necesară și i-l redau acestuia când secvența se termină.



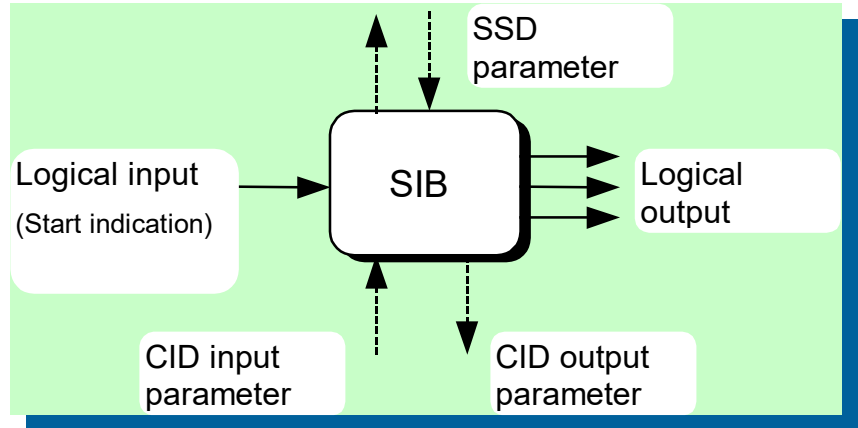
SIB și localizarea lor

Structura SIB

What is a SIB



Piece of tested SW with
- logical Inputs and
Outputs



CID = Call Instance Data (dynamische Parameter)
SSD = Service Support Data (statische Parameter aus der Service)

Un SIB este descris prin intermediul intrărilor, ieșirilor și operațiilor efectuate asupra acestora, cu excepția SIB-ului BCP care va fi descris în cele ce urmează. Intrările și ieșirile pot fi împărțite în trei clase distincte:

- Control

Fiecare SIB are un punct logic de start care indică locul de unde începe execuția și câteva puncte logice de oprire care îl includ întotdeauna pe cel de eroare. Opririle logice indică natura modului în care s-a terminat execuția și permit continuarea acesteia pentru diferite variante. De exemplu, SIB-ul ce efectuează verificarea are două terminații în afară de cea de eroare: verificare terminată cu succes sau verificare terminată fără succes (vezi [PIC2 Authorize Origination Attempt](#) din cadrul BCSM).

- Date de suport ale serviciilor (SSD = Service Support Data)

SSD reprezintă informații constante oferite atunci când un SIB este inițiat. SSD conține informații despre argumentele de intrare ale SIB-ului, precum tipul și locul în care este stocat. Tot ca date SSD sunt date și locația și tipul rezultatului. De asemenea pot fi oferite și valori constante, ca de exemplu formatul care este folosit de SIB-ul "verificare" pentru comparație.

- Date ale instanței apelului (CID = Call Instance Data)

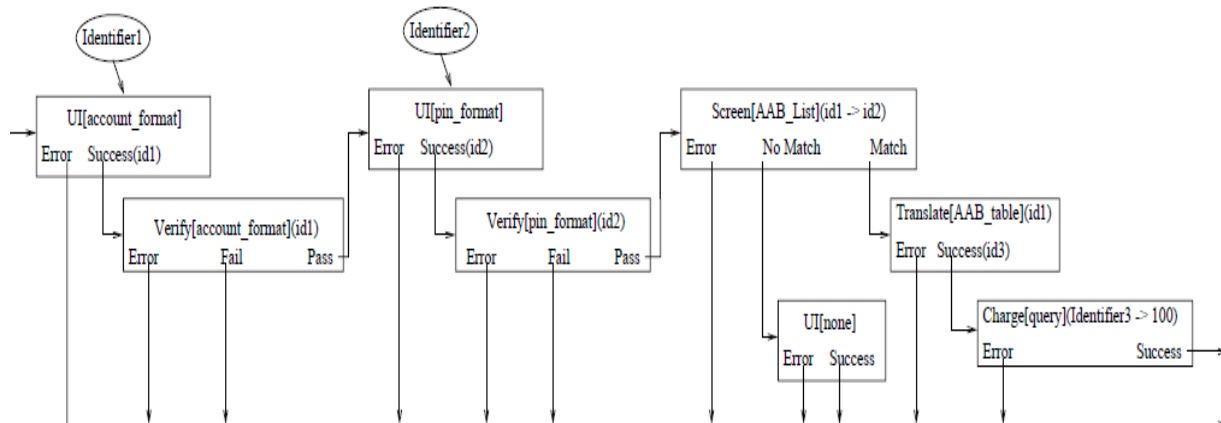
CID reprezintă datele dinamice și va fi accesibil SIB-ului în timpul execuției. Locul în care se găsesc datele dinamice este dat de pointeri statici. Un exemplu de date dinamice este informația pe care SIB-ul "verificare" o compară cu formatul.

Legătura dintre SIB-uri este specificată prin conectarea capetelor logice de intrare și ieșire, iar datele sunt trecute printr-un lanț de SIB-uri folosind locații comune pentru parametrii de intrare și pentru ieșiri. Aceasta este exemplificată grafic în figura de mai jos care detaliază serviciul de facturare alternativă automată (AAB = Automatic Alternative Billing service).

Serviciul AAB transferă facturarea unui apel către un alt cont, diferit de cel implicit. Pentru început utilizatorul este întrebă în legătură cu noul cont care să fie folosit și apoi îi este cerut PIN-ul. Atât contul cât și PIN-un au propriul format și sunt verificate înainte de a se trece la pasul următor

- Contul este corelat cu PIN-ul și se verifică dacă transferul este permis.
- Apoi noul cont este identificat folosind SIB-ul “Translate”.
- Datele contului sunt oferite împreună cu indicația ca acesta să fie folosit pentru taxare către SIB-ul “Charge” (taxare) care efectuează pașii necesari.

Fiecare dintre SIB-uri este reprezentat cu un dreptunghi în care numele este parametrizat cu numele instanței și datele de intrare. De exemplu, primul SIB din lanț (“Verify”) este reprezentat ca `Verify[account_format](id1)`. În acest caz, parametrul de inițiere – formatul contului (`account_format`) – informează SIB-ul despre faptul că trebuie să verifice formatul unui cont ce urmează a fi folosit, iar parametrul de intrare `id1` reprezintă locația în care se află datele ce urmează a fi verificate.



Serviciul de facturare alternativă (Automatic Alternative Billing System)

Direcția în care se face comunicarea este reprezentată de săgețile ce conectează SIB-urile. Controlul ieșirii trebuie etichetat cu stop logic, iar când este cazul, are și valori de ieșire date ca parametri. De exemplu, în figura de mai sus, SIB-ul “prima interacțiune cu utilizatorul” (UI = first User Interaction), în cazul unei execuții terminate cu succes va scrie în locația indicată de eticheta `Success(id1)` valoarea primită de la utilizator.

Transferul de date dintre SIB-uri este specificat prin intermediul unor nume simple de locații, precum `id1`, `id2`, și `id3` – ca în exemplul de mai sus.

Un SIB este compus dintr-un set de acțiuni ale entităților funcționale (FEAs - Functional Entity Actions) care fie independent fie în combinație cu alte SIB-uri sunt folosite ca entități de serviciu model.

SIBs sunt folosite de logica serviciilor și asigură o vedere globală asupra rețelei structurate IN ca și o singură entitate. SIB-urile sunt independente de distribuția capacităților în cadrul rețelei la nivel fizic.

Sunt reutilizabile și reflectă varietatea de capacități ce pot fi implementate în rețea. Ele sunt definite la nivel înalt, nu sunt implementate la nivel fizic, existând ca model abstract în nivelul funcțional global, la nivel fizic sunt mapate ca operații și proceduri. Astfel utilitatea lor se referă la un mod de compoziție, decompoziție la nivelul SCE (Service Creation Environment)

Următoarele SIBs sunt definite pentru IN CS-1:

1 – Algorithm

Aplică un algoritm matematic pe un set de date furnizând rezultatul.

2 – Charge

Acest SIB determină dacă există anumite elemente de tarifare speciale pentru apel, însemnând orice regulă adițională celor aplicate apelurilor de bază. În general implică identificarea:

- Resurselor pentru care apare tarifarea
- Structura de tarifare care se aplică pentru fiecare resursă;
- Unde sunt aceste tarife oglindite / redirecționate

Acest SIB nu este responsabil cu procedura de facturare pentru abonat, practic se ocupă numai cu aplicarea “charging-ului” pentru obținerea de informații ce vor fi folosite ulterior pentru facturare.

3 – Compare

Acesta compară un identificator cu o valoare de referință cu cele 3 rezultate posibile:

- Identificatorul este mai mare decât referința - GREATER THAN -
- Identificatorul este mai mic decât referința - LESS THAN -
- Identificatorul este egal cu referința - EQUAL TO -.

4 – Distribution

Permite utilizatorului să distribuie apelurile către diferite puncte terminale ale SIB dependent de parametri specificați de utilizator.

5 – Limit

Limitează numărul de apeluri corespunzătoare unei capacități a unui serviciu IN, bazându-se pe parametri specifici utilizatorului. Acest SIB nu este folosit pentru gestiunea congestiei în cadrul rețelei.

6 – Log Call Information

Creează log-uri pentru fiecare apel într-un fișier, utilizate mai târziu de către servicii de gestiune, ca exemplu statisticile, însă nu de către servicii ce țin de apeluri.

7 – Queue

Furnizează secvențierea apelurilor ce vor fi executate, către partea apelată.

8 – Screen

Execută o comparație a unui identificator cu o listă, pentru a determina dacă acesta este sau nu prezent în lista activă.

9 – Service Data Management

Activează modificări ale datelor specifice utilizatorului final, adaugare, înlocuire, interogare, incrementare sau decrementare.

10 – Status Notification

Acest SIB furnizează capacitatea de interogare a statusului și/sau schimbărilor în statusul resurselor rețelei.

11 – Translate

Determină informații din alte informații de ieșire după o prelucrare prealabilă.

12 – User Interaction

Permite schimbul de informații între rețea și o parte apelată sau apelantă.

13 – Verify

Acest SIB furnizează confirmarea dacă informația recepționată este consistentă sintactic respectiv dacă are forma informației așteptate.

14 – BCP (Basic Call Process)

Pentru procesul unui apel de bază în CS-1 s-a definit un SIB care gestionează aceste aspecte. Capabilitățile acestea permit folosirea logicii globale a serviciilor deasemenea și a altor SIBs pentru a descrie complet servicii CS-1 și capabilități subordonate acelor servicii.

Au fost specificate nouă puncte de inițiere (POI - Points of Initiation) pentru interfețele GSL (Global Service Logic) interfaces. În plus au fost specificate 6 puncte de returnare PORs (Points of Return). Astfel au fost definite:

POI

- Call Originated
- Address Collected
- Address Analysed
- Prepared to complete Call
- Busy
- No Answer
- Call Acceptance
- ActiveState
- End of Call

POR

- Continue with existing data
- Proceed with new data
- Handle as transit
- Clear Call
- Enable call party handling
- Initiate Call

SIB-ul BCP trebuie să implementeze din punct de vedere funcțional posibilitățile unui apel normal împreună cu cea de a decide când anume funcționalitatea IN este necesară.

Intenția lucrării de față nu este aceea de a modela în detaliu blocul BCP ci să ofere o bază de pornire pentru a dezvolta SIB-urile necesare simulării în întregime a rețelei inteligente. Procesul apelului de bază este descris la nivelul planului funcțional global doar sub forma unor secvențe de SIB-uri în care se poate urmări interacțiunea cu IN-ul. Interacțiunea este realizată prin descrierea interfeței dintre BCP și GSL (Global Service Logic) folosită la construcția secvențelor de SIB-uri. Interfața este alcătuită din puncte de inițiere (POI = Point of Initiation) și puncte de revenire (POR = Points of Return).

Punctele de inițiere sunt acele puncte din cadrul execuției BCP în care acesta poate detecta necesitatea funcționalității IN; atunci BCP este suspendat și controlul este preluat de GSL care inițiază o secvență de SIB-uri.

Punctele de revenire denotă diferite tipuri de a preda din nou controlul BCP-ului (via GSL) după ce o secvență de SIB-uri a fost executată.

Logica serviciilor

Logica serviciilor reprezintă un set de subrutine și reguli care interacționează cu capacitățile de rețea și cu modelul BCSM (Basic Call State Model) pentru a implementa servicii în cadrul rețelei structurate IN. Logica aceasta se împarte în 2 categorii Globală și Distribuită.

Logica globală (GSL) a fost definită ca “lanțul” care descrie ordinea în care se pot înșirui acele SIBs pentru a forma împreună servicii. Pentru un serviciu dat sau capacitate în cadrul unui serviciu GSL, ea este folosită pentru a descrie:

- Un POI specific care va defini punctul de lansare de la BCP către lanțul de SIBs
- Un set specific de POR unde lanțul logic SIB poate ceda controlul BCP (Basic Call Process)
- Formatul SIB-urilor și ordinea în care pot fi înșiruite
- Parametrii de configurare pentru fiecare SIB din lanț

Logica DSL (Distributed Service Logic) există în planul fizic (DFP). Există un set de logică DSL per SIB și folosește FAEs (Functional Entity Actions) și fluxuri informaționale pentru execuția serviciului.

Entitatea funcțională a unui apel/Modele de procesare în logica serviciilor

Modelele de procesare în logica serviciilor furnizează unelte folosite de proiectanții IN să modeleze apelurile să înțeleagă și să descrie distribuția de funcții între entitățile funcționale și comunicarea / relațiile dintre acestea.

Acestea includ procesarea și stabilirea conexiunii în cadrul apelului în SSF/CCF, execuția serviciului în SCF cât și folosirea resurselor și datelor din SRF, respectiv SDF.

Modelarea apelului furnizează o implementare abstractă la nivel înalt independentă de producător ce asigură o vedere pas cu pas a activităților în SSF/CCF și resurselor în SRF, oferind posibilitatea de interacțiune a SCF cu SSF în cursul execuției logicii serviciului.

Modelarea apelurilor pentru IN CS-1

Pentru a furniza o vedere pas cu pas în funcționalitatea SSF/CCF interacționând cu SCF modelul furnizează următoarele:

- un fundament bazat pe tehnologia din rețeaua existentă, în curs de dezvoltare
- o vedere asupra procesării SSF/CCF a apelului implicând ambele modele Originating și Terminating Basic Call State Models (BCSMs);
- un mediu pentru definirea necesităților pentru triggering pentru BCSM să invoce logica serviciului de pe IN și să raporteze evenimentele din timpul procesării apelului către logica serviciilor de pe IN respectiv DPs (Detection Points), mediu ce poate fi folosit de către proiectant pentru dezvoltarea serviciilor în cadrul rețelei;
- un nivel de aplicație / mediu pentru asigurarea secvențierii corecte a funcțiilor din SSF/CCF referindu-ne la cazul BCSM și anume PICs (Point In Calls) și tranziții
- reguli pentru reprezentarea și gestiunea interacțiunilor instanțelor de logică a serviciului
- o bază pentru definirea fluxurilor de informație dintre un SSF și un SCF

Exemple de funcții de procesare / stabilire conectare a apelurilor, către SCF de la SSF/CCF:

- influența în derularea apelului (ex. rerutare apel)
- acces și schimb de informație referitoare la procesarea apelului (ex. translatare adresă, informații de rutare)
- manipularea conectivității apelului (ex. redirectare apel - forwarding);
- monitorizare de evenimente legate de procesare apel și manipulare conectivitate (ex. no answer, busy, disconnect)

Modelarea procesării serviciului

Similar cu modelul oferit ca fundament sau mediu de dezvoltare, respectiv vedere abstractă asupra activităților la nivel SSF/CCF modelarea procesării serviciului oferă o

abstractizare independentă de produs a procesării la nivel SCF, respectiv activități SRF și SDF.

Se furnizează o modelare la nivel înalt, o abstractizare a procesării serviciului fără însă a specifica recomandări referitoare la implementarea în sine.

Relația între modele și produs

Modelarea apelurilor și a logicii serviciilor asigură numai o vedere de ansamblu asupra activităților și resurselor a SSF/CCF, SCF, SDF, SRF, aceste modele nu implică obligativitatea ca producătorii să proiecteze entități funcționale în produsele proprii ca mapari unu la unu a componentelor funcționale ale modelelor.