HB 1423

HÍRADÁSTECHNIKA

XLIII. ÉVFOLYAM 1992. AUGUSZTUS

TÁVKÖZLÉSI SZOFTVEREK

A szerkesztő gondolatai ..... Csopaki Gy. 1

A hálózat-üzemvitel szabványosításának helyezte ..... H. Ségot, C. Lecerf, P. Joly 2

Protokollok specifikálása és tesztelése ..... Tarnay K. 9

Az AXE 10 szoftvertervezési folyamatban alkalmazott technológia az Ericssonnál ..... S. Kotlo 21

Az objektum-orientált programozási technika alkalmazása távközlési szoftverek előállítására .. Adamis G. 26

Szoftverfejlesztés elosztott rendszerek számára ..... Valló F. 30

Gazdaság – Kutatás – Oktatás

A magyar távközlést érintő nemzetközi tevékenységről ..... Villányi O. 33

Hírek – Események

Könyvismertetések 38

Ferenczy Pál (1931–1992) ..... Barát Z. 39

HÍRADÁSTECHNIKA
A HÍRADÁSTECHNIKAI TUDOMÁNYOS EGYESÜLET FOLYÓIRATA

Főszerkesztő
BARANYI ANDRÁS

Rovatvezetők
BATTISTIG GYÖRGY
KORMÁNY TERÉZ
PRÓNAY GÁBOR
SOMOGYI ANDRÁS

Szerkesztők
BARTOLITS ISTVÁN
KÁSA ISTVÁN
LADVÁNSZKY JÁNOS
FÖLDVÁRINÉ OROSZ JULIANNA
ANTALNÉ ZÁKONYI MAGDOLNA

Munkatárs
ANGYAL LÁSZLÓ

Szerkesztőbizottság
TÓFALVI GYULA
elnök
BERCELI TIBOR
FRAJKA BÉLA
FRIGYES ISTVÁN
GORDOS GÉZA
MOJZES IMRE
PAP LÁSZLÓ
SALLAI GYULA

SZPONZOROK

MAGYAR TÁVKÖZLÉSI RÉSZVÉNYTÁRSASÁG

MAGYAR MŰSORSZÓRÓ VÁLLALAT

ERICSSON
Ericsson Technika Kft.

MOTOROLA

Siemens Telefongyár Kft.

TKI

BHG BUDAPEST

KONTRAX TELEKOM

"AZ ÉPÍTÉS FEJLŐDÉSÉÉRT" ALAPÍTVÁNY

Szerkesztőség
Budapest II., Gábor Áron u. 65.
Postacím
1525 Budapest, Pf. 15.
Telefon: 135-1097
201-7471
Telefax: 135-5560

Előfizetési díj
Hazai előfizetők részére
1 évre 2900,– Ft, egyes számok 360,– Ft
HTE egyéni tagjai részére
1 évre 480,– Ft, egyes számok 60,– Ft
Külföldi előfizetők részére
1 évre 6 angol szám 60 USD, 12 szám 90 USD, egyes számok 15 USD

HÍRADÁSTECHNIKA megjelenik havonta váltakozva magyar és angol nyelven. Kiadja a TypoTEX Elektronikus Kiadó Kft. Budapest I., Batthyány u. 14. 1015. Telefon: 202-1365. Felelős kiadó: Votisky Zsuzsa. Készült a HUNGAPRINT nyomdában. Szövegszedés: TypoTEX Kft. A lap példányonként megvásárolható a nagyobb könyvesboltokban és a kiadónál.

HU ISSN 0018-2028

A SZERKESZTŐ GONDOLATAI

A távközlés digitalizálása nagy kihívást jelentett a távközlési területen dolgozó szoftverfejlesztők számára. A távközlési szoftvereknek a következő kövtelményeknek kell eleget tenniök:

- nagy sebességű real-time folyamatok hatékony és gyors megvalósítása,
- számos párhuzamos folyamat hatékony kezelése,
- nagy megbízhatóság,
- tesztelhetőség és diagnosztizálhatóság,
- igény szerinti paraméterezhetőség.

A számítástechnikai szoftvernek nincs hibatűrése, a távközlési szoftverek esetén azonban adott határon belüli tolerancia megengedett. Különleges követelményt jelent továbbá a távközlési szoftverek esetén a teljes folyamatos üzem, megállásra nincs lehetőség.

A távközlési szoftvereknek nagy hibatűrési lehetőséggel kell rendelkezni, mivel nagy számú felhasználó váratlan beavatkozásaira is fel kell készülni. A világméretű távközlési hálózat együttműködése megkötöttséget jelent a fejlesztők számára.

Az intelligens hálózatok megjelenésével a felhasználónak lehetőséget kell nyújtani a szolgáltatások programozására. A vázolt követelményeket kielégítő távközlési szoftverek többnyire az alkalmazott processzorok gépközeli nyelvén készültek. Nagyméretű hatékony programrendszerek készítése azonban manapság már hatékony formális leíráson alapuló módszerek nélkül igen nehézkes.

A folyóirat tematikus különszáma átfogja a távközlési szoftverek formális nyelvű specifikációjának témakörét, áttekinti az implementáció során alkalmazott objektumorientált távközlési szoftverfejlesztési technológiát. Külön kiemelt jelentőségű a digitális távközlési rendszerek protokolljainak technológiája, különös tekintettel e protokollok specifikációjára és az e leíráson alapuló szimulációjára, tesztelésére, verifikációjára és validációjára.

A távközlési hálózatok menedzselésének és hatékony működtetésének a megvalósítása a szolgáltatást nyújtó alapvető érdeke.

Magyarországon jelenleg a távközlés fejlesztése nagyon erős ütemben folyik a modern digitális távközlési rendszerelemek felhasználásával. Mint ahogyan a Magyar Távközlési Rt. folyóirata, a Magyar Távközlés is foglalkozott a távközlési szoftverek témakörével, e tematikus számnak az a célja, hogy szélesebb spektrumban, nem csak magyar szerzők írják le gondolataikat, illetve számoljanak be új szakmai és tudományos eredményeikről.

CSOPAKI GYULA

Csopaki Gyula a BME Villamosmérnöki Karán végzett 1969-ben. Először az MTA Informatikai és Elektronikai Tanszéki Kutatócsoport munkatársaként dolgozott 1987-ig, majd a Híradástechnikai Elektronika Intézet, jelenleg pedig annak egyik jogutódjának, a Távközlési és Telematikai Tanszéknek a docense. A műszaki tudományok kandidátusa fokozatot 1989-ben szerezte a digitális rendszerek formális nyelveinek és az azokon alapuló módszerek témakörben. Jelenleg a távközlési szoftverek magas szintű formális nyelvi eszközeivel és azok megvalósítási kérdéseivel foglalkozik digitális távközlési rendszerek számára. Tagja az MTA Távközlési Rendszerek Bizottságának, HTE-nek, MATE-nek, NJSZT-nek, MMÉVSZ-nek.

A HÁLÓZAT-ÜZEMVITEL SZABVÁNYOSÍTÁSÁNAK HELYZETE

HUBERT SÉGOT
CNET/PAA/RGE, 38-40, RUE DU GÉNÉRAL-LECLERC
F-92131 ISSY-LES-MULINEAUX

CHRISTIAN LECERF
CNET/LAA/SLC, ROUTE DE TRÉGASTEL, BP 40
F-22301 LANNION CEDEX

PASCALE JOLY
CNET/PAA/RGE, 38-40, RUE DU GÉNÉRAL-LECLERC
F-92131 ISSY-LES-MULINEAUX

Napjainkban igen gyorsan növekszik a különböző információfajták (beszéd, adat, kép) átvitele iránti igény, ami a nyilvános és magáncélú hálózatok elburjánzásához vezet. E hálózatoknak — sokféleségük ellenére — együtt kell működniük. Az ilyen bonyolult és létfontosságú hálózatokat az üzemvitel jó megszervezésével lehet kézben tartani. Emiatt és a felhasználók nyomásának hatására a hálózat-üzemvitel (és főleg ennek szabványosítása) manapság teljes pezsgésben van. E cikk a hálózat-üzemvitel területén végbement fejlődés jelenlegi állapotát mutatja be a fontosabb szabványügyi szervezetek keretein belül, arra törekedve, hogy kiemelje a főbb irányzatokat. Szólunk arról is, milyen tapasztalatokat szerzett a France Télécom a szabványok alkalmazásában.

1. BEVEZETÉS

A távközlési eszközök iránti igény napjainkban igen gyorsan növekszik. A General Motors leányvállalata, az Electronic Data System üzemelteti például jelenleg a világ egyik legnagyobb magánhálózatát. A nemzetközi hálózat naponta 25 milliárd tranzakciót bonyolít le, a növekedés mértéke évenként 300 %-os. Hozhatnánk példának a francia telefonhálózatot is, ahol a helyközi forgalom hatévente, a nemzetközi forgalom pedig négyévente megkétszereződik.

Egyébiránt az igények egyre összetettebbé válnak, és a különböző információfajták (beszéd, adat, kép) átvitele iránti igény a nyilvános és magáncélú hálózatok elburjánzásához vezet. E hálózatoknak — sokféleségük ellenére — együtt kell működniük.

Az információcsere a vállalkozás meghatározó elemévé vált, ami oda vezetett, hogy a vállalatok erősen függővé váltak az említett hálózatok működésétől.

Az ilyen bonyolult és létfontosságú hálózatokat az üzemvitel jó megszervezésével lehet kézben tartani. Emiatt és a felhasználók nyomásának hatására a hálózat-üzemvitel (és főleg ennek szabványosítása) jelenleg pezsgésben van: egyedül az Egyesült Államok magánhálózatai üzemviteli rendszereinek piaca 427 millió dollárt jelentett 1989-ben, 1996-ra pedig 2,5 milliárd dollárt becsülnek.

Ami a nyilvános hálózatokat illeti, ott a rövid távú gondolkodás oda vezetett, hogy nem alakulhatott ki a hálózat-üzemvitel tényleges politikája. Rövid távon hasznosnak bizonyul az a sokfajta berendezés, amelyet azért helyeztek üzembe, hogy megoldják a hálózat egy pontján jelentkező gondokat, elterjedésük hosszú távon azonban katasztrofális eredménnyel jár. Nem ritkán egész termeket látunk, olyan egyéni rendszerű adat-végberendezésekkel telezsúfolva, amelyek mindegyike egy-egy rendszerhez tartozik.

A magánhálózatok üzemeltetőinek legfőbb gondja a berendezések sokféleségéből adódik: minden egyes berendezésgyártó a saját, egyedi üzemviteli rendszerét kényszeríti rá a vásárlóra (az Egyesült Államokban egy átlagos magánhálózat 20 különböző gyártó által szállított berendezésből épül fel) [1]. A vállalatok manapság általában nem rendelkeznek hálózatukról átfogó, központosított képpel.

A magáncélú és nyilvános hálózat üzemeltetőinek törekvése itt tehát találkozik, összeköti őket egyrészt a szabványosítás fejlődése (open network provision, open network architecture), másrészt a technika növekvő bonyolultsága. Mindketten igénylik azoknak a szabványoknak a megjelenését, amelyek függetlenítik őket a gyártóktól, lehetővé téve, hogy saját hálózatuk üzemviteli rendszere jövőálló legyen.

Jelen cikk a különböző szabványügyi szervezetek által végzett munka jelenlegi állását mutatja be, egyúttal arra törekszik, hogy kiemelje a főbb szempontokat és azok kapcsolatait.

Először röviden ismertetjük a hálózatok üzemvitelének lényegét, a kapcsolódó területeket, majd bemutatjuk az ISO és a CCITT keretén belül végzett szabványosítási munka jelenlegi állását.

2. MIT JELENT A HÁLÓZAT ÜZEMVITELE?

A hálózat-üzemvitel azoknak a tevékenységeknek az összeségét jelenti, amelyek egy távközlési hálózat erőforrásainak felhasználását és működtetését ellenőrzik, felügyelik, rögzítik és tervezik annak érdekében, hogy a felhasználók a távközlési szolgáltatásokat a megkívánt színvonalon vehessék igénybe.

Megjegyezzük, hogy a „hálózatigazgatás" kifejezést gyakran ugyanabban az értelemben használják, mint amilyenben mi e cikkben a „hálózat-üzemvitel" kifejezést; tulajdonképpen a „network management" két különböző fordításáról van szó.

Az ISO és a CCITT a hálózat-üzemvitelnek következő öt területét különbözteti meg:

- a hálózat kialakítása, amely magában foglalja a szerelés, az üzembe helyezés, a működtetés adatainak kezelését (a fizikai és logikai erőforrások állapotát);
- a hibák kezelése, ami magában foglalja a karbantartást, különösen a riasztás általi felügyeletet, a hibabehatárolást és a vizsgálatokat;
- a teljesítmény figyelemmel kísérése: forgalomvezérlés, a hálózat terhelésének korkátozása (a forgalom mérése, figyelése, hálózatszervezési parancsok), a szolgáltatás minőségének ellenőrzése;
- tarifikáció, vagyis a számlák gyűjtése, a számlázási változóinak beállítása, valamint általában a hálózat erőforrásai használatának mérésével összefüggő valamennyi feladat;

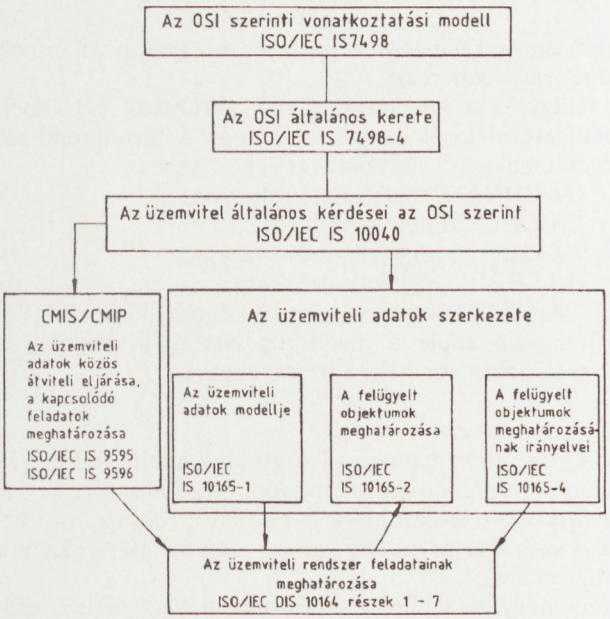
- a biztonság megteremtése, annak érdekében, hogy védjük a rendszer adatait és szolgáltatásait a jogtalan hozzáférés ellen (jogosítás, felhatalmazás), valamint védjük a rendszert a hibás működés ellen (a berendezések tartalékolása, dinamikus átkapcsolása hiba esetén stb.).

A hálózat-üzemvitelt ugyanakkor két szemszögből, az idő és a szolgáltatás szemszögéből kell vizsgálnunk. A hálózat-üzemviteli rendszer megvalósítása során az időtényező egyike a legalapvetőbb követelményeknek: 20 év távlattal tervezni a hálózatot teljesen más kérdéstípust vet fel, megoldásához teljesen más eszközök kellenek (kevés párbeszéd, elvont matematikai modellek alkalmazása), mint a hálózatfelépítés valós idejű szervezése, amely inkább arra irányul, hogy az átalakításokkal a forgalmat védje (nagy mennyiségű adat folyamatos cseréje a berendezésekkel, előre megadott hálózatátalakítási szabályok alkalmazása).

3. AZ OSI ÜZEMVITELI MODELL

A távközlési környezet (open system interconnection) szabványosítása után az ISO az üzemvitel szabványosítását vette célba, s ez nem más, mint az üzemviteli adatok átvitele nyílt rendszerek között.

Az 1. ábra az OSI üzemviteli modell egyes szabványait és azok kapcsolatát mutatja.



1. ábra. OSI-szabványok a hálózat üzemvitelében

Az OSI-környezetben az üzemvitel két területre bomlik:

- *Az üzemviteli adatok modellezése*

A cél: szabványosítani OSI-környezetben a nyílt rendszerek között átvitt üzemviteli adatokat; ezek az adatok a rendszer OSI-környezetének elvont megjelenítéseként foghatók fel.

- *Az üzemviteli adatok átvitele*

Ez a terület az üzemviteli adatok átvitelét szolgáló feladatok és átviteli eljárások szabványosítását foglalja magában.

3.1. Az adatok modellezése [ISO 10164 1, 2 és 4 jelű dokumentum]

A modellezés célja, hogy az üzemvitel szempontjait figyelembe véve meghatározzuk a felügyelni kívánt erőforrások elvont modelljét akár logikai (programrendszer eleme, hívásirányítás, irány...), akár fizikai (berendezés, központi egység, áramköri kártya, kábel...) erőforrásokról van szó. Az ISO a modell megalkotása során olyan megközelítést választott, amelyben egy valós erőforrást egy objektum modellez (felügyelt objektum). Az objektumot névvel megjelölt olyan független egységnek tekinti, amelyen műveleteket lehet végezni.

A felügyelt objektumot az alábbi tulajdonságok együttese jellemzi:

- az objektumon értelmezett műveletek összessége,
- az objektum által kezelt adatoknak megfelelő jellemzők, amelyeket csak műveletek segítségével lehet megváltoztatni,
- a meghatározott esemény bekövetkeztekor az objektum által küldött üzenetek.

Az azonos tulajdonságok együttesével rendelkező objektumok azonos osztályba tartoznak.

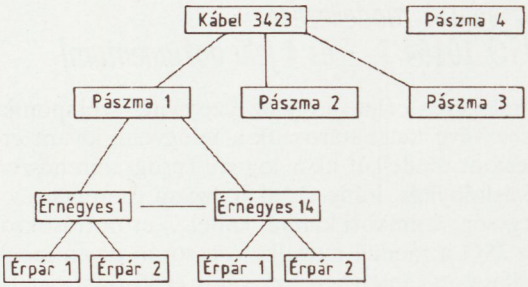
Az értékadás segítségével a saját osztály meghatározásából kiindulva és a jellemzőknek értéket adva (esetleg átmásolva ugyanannak az osztálynak egy létező objektumát) új felügyelt objektumot lehet létrehozni (2. ábra).

	« Értékadás »
előfizető	OBJEKTUM
OBJEKTUMOSZTÁLY	
JELLEMZŐK	
azonosító	azonosító=AS1234JKL89009654
név	név=DUPONT Jean
cím	cím=10 rue de la Paix,PÁRIZS
telefonszám	telefonszám=44456789
készüléktípus	készüléktípus=analóg
készülékszám	készülékszám=9876
szerződésszám	szerződésszám=XZ3200
TEVÉKENYSÉG	
áthelyezés	
ÜZENET	
szolgáltatás használatára jogtalan	

2. ábra. Értékadás

A felügyelt objektum további objektumot tartalmazhat. A távkábelt például olyan felügyelt objektumnak tekintjük, amely tartalmazza a „pászma" objektumokat, ez utóbbiak tartalmazzák az „érnégyes" objektumokat, amelyek maguk is tartalmazzák az „érpár" objektumokat. Az egymásba ágyazódás viszonyait a beágyazódási fa jellemzi, melynek segítségével felsorolhatjuk az objektumokat (3. ábra).

Íly módon minden objektum rendelkezik a relatív név (relative distinguished name) jellemzővel. Ez különbözteti meg őt a beágyazódási fa azonos szintjén lévő más objektumoktól. A tekintett tartományon belül minden objektum rendelkezik megkülönböztető névvel (distinguished name), amely a tartomány objektumait különbözteti meg egymástól. Példánkban az összes érpár közül kiválasztott egyetlen érpár megkülönböztető nevét úgy kapjuk meg, ha láncba fűzzük az adott érpár relatív nevét és az érpárat tartalmazó érnégyes, pászma, kábel relatív nevét. (Tehát: a kábelek tartományában 2. érpár, 5. érnégyes, 3. pászma, 3423. kábel a megkülönböztető neve egy adott érpárnak. Ugyanannak az érpárnak a relatív neve: 2. érpár.)



3. ábra. Beágyazódási fa. Példa: 112 érpárt tartalmazó kábel

Az objektumorientált modellben az öröklés mechanizmusa lehetővé teszi, hogy egy adott osztály eleme egy másik osztály tulajdonságait örökölje (4. ábra).

« ÖRÖKLÉS »

előfizető	ISDN-előfizető
OBJEKTUMOSZTÁLY	OBJEKTUMOSZTÁLY
JELLEMZŐK	
azonosító	az előfizető osztályból
név	ÖRÖKLI
cím	
telefonszám	JELLEMZŐK
készüléktípus	D-csatorna típusa
készülékszám	B-csatornák száma
szerződésszám	alácímzés
TEVÉKENYSÉG	
áthelyezés	
ÜZENET	
szolgáltatás használatára jogtalan	

4. ábra. Az öröklés

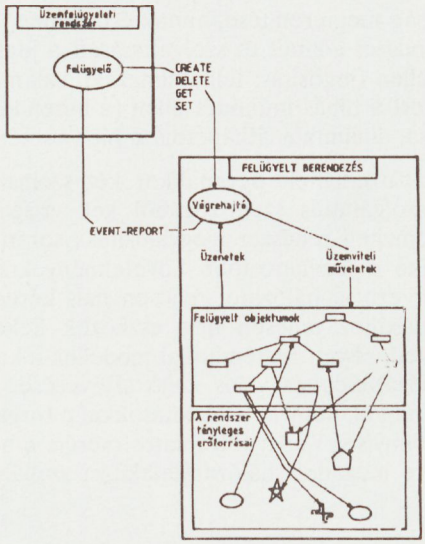
Az öröklés lehetővé teszi, hogy osztályozzuk az objektumokat, és ezzel elkerüljük az átfedéseket az osztályok kijelelölésénél. Általános objektumosztályokat hozhatunk létre, amelyekből kiindulva öröklés segítségével szűkebb osztályokat lehet kijelölni. Felvehetjük például az általános előfizető osztályt (amely tartalmazza a nevet, a címet, a szerződésszámot...), majd létrehozzuk a telefonelőfizető, ISDN-előfizető, rádiótelefon-előfizető osztályokat, amelyek öröklik az előfizető osztály tulajdonságait, továbbá tartalmazzák az adott osztályra jellemző tulajdonságokat is.

Az öröklés a modell továbbfejlesztését is megkönnyíti, hiszen fel lehet használni a meglévő osztályokat. Példa erre az új „szélessávú" előfizetői típus bevezetése.

Fontosnak tartjuk megjegyezni, hogy az objektumorientált modell több, mint az adatok puszta modellezése: lehetővé teszi, hogy kifejezzük az általa leírt erőforrások időbeli változását. Az objektum a műveletek révén feladatok elvégzésére is képes, tehát rendelkezik szolgáltatási képességgel is. A szolgáltatásokat a viselkedése adja.

3.2. Az üzemviteli adatok átvitele (CMIS, CMIP: ISO 9595 és 9596 jelű dokumentum)

Az adatátviteli szolgáltatások (CMIS, common management information service) és eljárás (CMIP, common management information protocol) segítségével egyrészt egy rendszer objektumain (és ezzel egyúttal az általuk modellezett erőforrásokon) távolról lehet műveleteket végezni, másrészt továbbítani lehet az objektumok által küldött üzeneteket (5. ábra).



5. ábra. A központosított felügyelet szolgáltatásai és átviteli eljárása: CMIS/CMIP

Az üzemviteli információk CMIS szolgáltatások segítségével történő továbbítása két folyamat közbejöttét tételezi fel, amelyek a közvetítés szempontjából az alábbi két szerep egyikét játszhatják:

A felügyelő szerepe:

- Az üzemvitel oldaláról a rendszer felügyelt objektumok képében jelenik meg;
- a felügyelő a következő CMIS parancsok egyikének kiküldésével kezdeményezheti, hogy a távoli rendszer objektumai adott műveleteket végezzenek el:
  - M-CREATEobjektum létrehozása,
  - M-DELETEobjektum törlése,
  - M-GET egy jellemző olvasása,
  - M-SET egy jellemző írása,
  - M-ACTION egy tevékenység végrehajtása;
- a felügyelő kapja a távoli rendszer által küldött, az objektumokra vonatkozó üzeneteket.

A végrehajtó szerepe:

- A végrehajtó kapja meg a felügyelőtől a felügyelt objektumon elvégzendő műveletre vonatkozó paran ot;
- az ő feladata lefordítani a parancsot a rendsz erőforrásai számára, és esetleg választ vagy nyugtát küldeni a felügyelőnek;
- kezdeményezi a rendszer erőforrásai felől érkező eseményjelentés lefordítását objektumorientált üzenetté és továbbítja ez utóbbit az felügyelőnek;
- a továbbításhoz CMIS M–EVENT–REPORT szolgáltatását használja.

3.3. Az üzemfelügyeleti rendszerszolgáltatások (SMF) elemei

Az egyes tartományok feladatainak elemzése nyilvánvalóvá tette, hogy van néhány olyan „támogató szolgáltatás"*, amely minden tartományban megegyezik. Jellemző

* A támogató szolgáltatás, más néven az üzemfelügyeleti rendszerszolgáltatás angol elnevezésének (system management function) rövidítése SMF, amelyet a továbbiakban mi is megtartunk, lévén alkalmazása következetes.

nyítását, illetve az események naplózását. Az SMF szolgáltatásai egymást is hívhatják, például a riasztások feldolgozó programja (alarm reporting) használja a naplózóprogramot. Az építmény alappillére az objektumkezelés (object management), amelyet az összes többi program használ, és amely tulajdonképpen közvetlenül a CMIS szolgáltatásainak felel meg (CMIS teszi lehetővé, hogy távolról meghívjuk az objektumkezelés szolgáltatásait).

A szolgáltatások és az átviteli eljárások technikájának szabványosítása az utolsó fázisban van. Az ISO jelenleg az üzemfelügyeleti rendszerszolgáltatások szabványosítására összpontosít. A készülő szabvány nyitott abban az értelemben, hogy szükség esetén további szolgáltatásokat lehet majd szabványosítani (jelenleg 19 szolgáltatás szabványosításán dolgoznak).

Az objektumorientált megközelítés egyik jellemzője, hogy az objektum fogalma egyesíti magában egyrészt a statikus aspektusokat (állapotok leírása jellemzőik segítségével), másrészt — az objektumokon végrehajtható műveleteken keresztül — a működésbeli aspektusokat (szolgáltatásokat). Ilyen megközelítésben az üzemfelügyeleti rendszerszolgáltatás olyan műveletnek tekinthető, amelyet az objektumok kínálnak fel, és az SMF szabványok ténylegesen tartalmazzák is egy vagy több olyan objektum leírását, amelyek működése megfelel a kívánt szolgáltatásnak. Például: Az eseményfeldolgozást leíró anyag tartalmazza az osztályozó objektumosztály meghatározását. Az üzemfelügyelet létrehozhat egy olyan osztályozó objektumot, amely egy adott eseményt jelző üzenetet a feladás időpontja vagy egyes váltózók értékének függvényében rendez, majd a rendezett üzenetet továbbirányítja az előre megadott címzetthez.

A naplózóprogramot leíró anyag tartalmazza a napló objektumosztály meghatározását. Az üzemfelügyelet létrehozhat egy olyan napló objektumot, amely egy adott eseményt jelentő üzenetet tárol.

Az üzemfelügyeleti rendszerszolgáltatás elemeit (pl. osztályozó létrehozása, törlése, létező osztályozó működési szempontjainak módosítása...) tehát úgy tekinthetjük, mint az objektumkezelés (create, delete, set...) alkalmazását egy meghatározott objektumra (az osztályozóra).

Az SMF következő alapelveinek szabványosítása van folyamatban: objektumkezelés, állapotátmenet-kezelés, az objektumok egymás közötti viszonya, riasztások, eseményfeldolgozás, naplózás, biztonsági riasztás, hozzáférés-védelem, költségszámítás, terhelés-ellenőrzés, vizsgálóeljárások, időgazdálkodás stb.

Megjegyezzük, hogy a szabványosítás alatt álló felületek használják a számítási és átviteli erőforrásokat. Az ebből adódó többletterhelés biztosan nem okoz gondot az erőforrásokkal gazdagon ellátott berendezések esetében, viszont válságos helyzetet okozhat a jelenlegi, szerényebben méretezett berendezésekben. Ugyanakkor az erőforrások olcsóbbá válása és az igazgatások azon törekvése, hogy függetlenné válljanak a gyártóktól és a „tulajdonosi felületektől" az ilyen típusú felületek egyre következetesebb alkalmazása felé mutat.

4. A TÁVKÖZLÉSI ÜZEMVITELI HÁLÓZAT

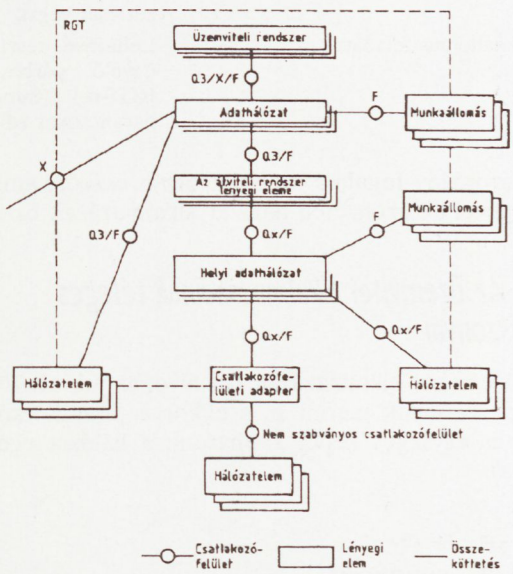
A CCIT az ISO-ét kiegészítő megközelítést választott: az OSI szabványait a távközlés szükségleteinek megfelelően egészíti ki.

A távközlési hálózatok szerkezetének fejlődése jelenleg a „hálózati alrendszer" és a „üzemviteli alrendszer" világos működésbeli elkülönülése felé mutat. a CCITT kidolgozta az említett üzemviteli alrendszer szerkezeti kereteit, ez az RGT (távközlési üzemviteli hálózat, angolul telecommunications management network).

4.1. Az RGT felépítése

Az RGT lényegi elemeit az 1. táblázat adja meg.

Vonatkoztatási pontnak tekintjük azokat a pontokat, ahol az egyes működési egységek között az információ áthalad. A vonatkoztatási pontok akkor válnak csatlakozófelületté, ha az egyes működési egységeket különböző berendezések valósítják meg.



6. ábra. Az RGT fizikai felépítésének példája. (Megjegyezzük, hogy kétféle Q illesztőfelület képzelhető el: Q_3 és Q_x. Q_x Q_3 részhalmazát képezi, és olyan berendezésekben használatos, amelyek vagy kevés számítási erőforrással rendelkeznek, vagy nincs szükségük Q_3 valamennyi szolgáltatására.)

A 6. ábra a rendszer szerkezeti felépítését mutatja, különös tekintettel az egyes csatlakozófelületek (Q, F, X) elhelyezkedésére.

A felületek használják a CMIS szolgáltatásokat és a CMIP átviteli eljárást (l. a 3.2. fejezetben); használhatják még az FTAM (file transfer and access management) adatállomány-átviteli eljárást is. A felülethez kapcsolódó objektumok meghatározása gyorsan halad.

4.2. Az üzemvitel gyakorlati alkalmazása

4.2.1. A tartomány fogalma

Az üzemvitel szervezése tekintetében a felügyelt objektumokat különböző (működésbeli, földrajzi, felépítésbeli...) szempontok alapján tartományokra osztjuk. A tartományok lehetnek átlapolódók vagy nem átlapolódók. A felosztás az idővel változhat: például egy adott tartományért felelős üzemviteli központ időlegesen átadhatja feladatai egy részét egy másik központnak, mondjuk, meghibásodás vagy munkaidőn túli zárva tartás miatt.

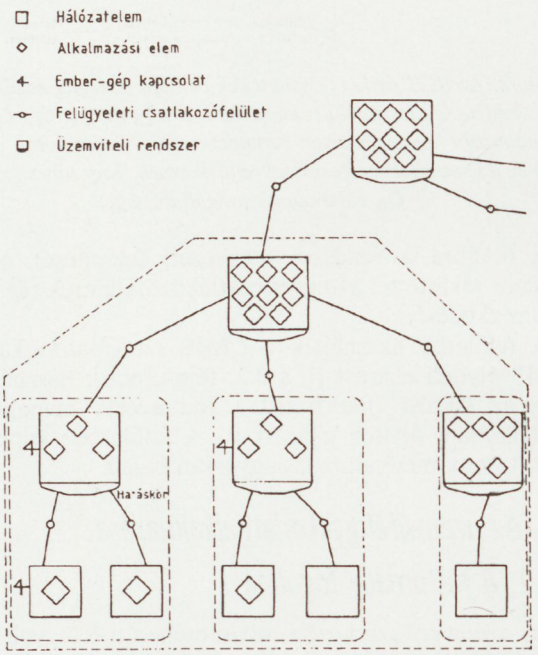
1. táblázat. Az RGT lényegi elemei

Feladat	Meghatározás	A megfelelő lényegi elem
Az üzemviteli rendszer feladata	Feldolgozza az üzemviteli információkat. Vezérli és felügyeli a berendezéseket, figyelembe veszi az ember által adott esetleges parancsokat is.	Az üzemviteli rendszer
Az átviteli rendszer és a *Q* felület feladata	Az egyik adatábrázolási rendszerről egy másikra való fordítást végzi (tulajdonképpen megváltoztatja az üzenetek tartalmát). Más szóval: „szintre hozza" az elavult vagy kevés számítási erőforrással rendelkező berendezéseket.	Az átviteli rendszer lényegi eleme *Q*-felületi adapter
A hálózatelem feladata	Az információ (beszéd, kép, adat) továbbítását, a helyi felügyeletet és a feladatvégrehajtás helyi vezérlését végzi.	A hálózatelem
A munkaállomás feladata	Lehetővé teszi, hogy a hús-vér felügyelő párbeszédet folytasson az RGT-vel (információkat kapjon és parancsokat adjon).	A munkaállomás

A tartomány fogalma az egyik olyan eszköz, amelynek segítségével az üzemviteli hálózat alkalmazásait osztályozhatjuk.

4.2.2. Az üzemvitel alkalmazásainak réteges viszonyai

Az üzemvitel alkalmazásait osztályozhatjuk alá-fölérendeltségi viszonyuk szerint is, s ekkor a piramis csúcsáról átfogó és egységes képet kaphatunk a hálózat egészéről (7. ábra).



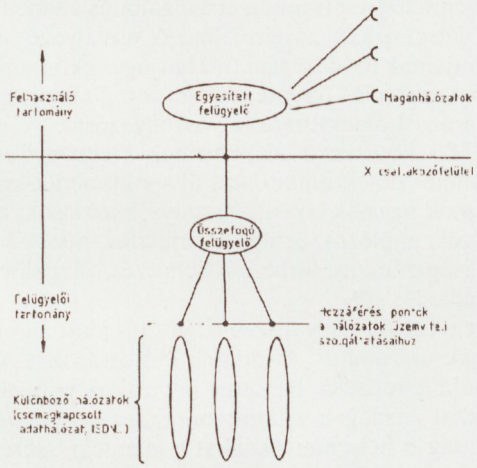
7. ábra. Réteges szerkezeti felépítés

Minden egyes közbenső rétegből más-más képet kapunk a hálózat egy részhalmazáról vagy az azonos típusú hálózatelemek összeségéről. Az alsó síkon a hálózatelemek találhatók. Minél feljebb haladunk a piramisban, annál átfogóbb képet kapunk.

A rendszer megvalósításának jelenlegi gyakorlatában a következő négy fő réteget szokás megkülönböztetni: a hálózatelemek felügyelete (a tevékenység tervezése, a végrehajtás nyugtáinak elkészítése...), a hálózat-üzemvitel (irányítási kérdések), a szolgáltatások nyújtása (a szolgáltatások minősége, az előfizetők nyilvántartása...), vállalatvezetés (kereskedelmi és jogi szempontok, piacelemzés...).

4.2.3. Felhasználói hozzáférés

Az RGT magában foglal egy olyan üzemviteli szolgáltatást is, amely lehetővé teszi, hogy egyetlen felügyelő* összehangolt képet kapjon különböző nyilvános hálózatokról.



8. ábra. Példa arra, milyen felépítésű lehet a felhasználónak az RGT-hez történő szerves hozzáférése. Az ilyen „ablakon" keresztül a felhasználó az RGT-ben adatokat olvashat és parancsokat adhat, ezért döntővé válik a biztonság kérdése.

Az üzemviteli szolgáltatás fő célja a következő:
- több RGT hozzáférési pontjának összefogása,
- esetleges kapcsolatok megvalósítása,

* ha ezt elkülönült lényegi egységgel valósítjuk meg, akkor a neve: összefogó felügyelő

- értéknövelt szolgáltatások (statisztika, részletes számlázás, parancsok programozása stb.) nyújtása,
- különböző hálózatokról összefoglaló kép alkotása.

Tájékoztatásul megjegyezzük, hogy a felhasználó tartományában elképzelhető egy olyan, egyesített felügyelőnek nevezett lényegi egység, amely a felhasználó hálózata nyilvános és zártcélú részének felügyeletét egyesíti magában (8. ábra).

5. AZ OSI ÜZEMFELÜGYELETI ELKÉPZELÉSEINEK KIPRÓBÁLÁSA

5.1. Bigaroo [2], [3]

A hálózat üzemvitelének és a távközlési szolgáltatásoknak ezt a modelljét 1990-ben valósították meg a CNET Lannion A központjának SLC/LCM osztályán, hogy segítségével tapasztalatokat szerezzenek az objektumok alkalmazásáról.

A Bigaroo modellt Ethernet-hálózatra csatlakozó Sun munkaállomásokon valósították meg, és a következő négy alkalmazási blokkból állt: a felhasználó-hálózat párbeszédet megvalósító felület, a felhasználói végberendezések, a távközlési és a felügyeleti gazdagép. A modell által kezelt forgalmi minta többek között tartalmazza a széles sávú multimédia szolgáltatást [3], melynek összetevői változtathatók: a végberendezésekről lehet őket aktiválni. Ez a programrész az Eiffel objektumorientált nyelven íródott (a 150 objektum leírása körülbelül 30 000 Eiffel nyelvű forrássornak felel meg).

Az egyik cél az volt, hogy az OSI üzemviteli rendszerét úgy alkalmazzák a Bigaroo szolgáltatások kezelésére, hogy az üzemvitelnek a lehető legkevesebb hatása legyen magára a rendszerre.

A Bigaroo üzemvitele [3] három síkot tartalmaz: a) az üzemviteli központ, melynek feladata az ember–gép párbeszéd lehetővé tétele; b) az elosztott üzemviteli rendszer, melynek az a feladata, hogy a parancsokat elemezze és az egyes hálózatelemek felé továbbítsa, valamint hogy a hálózatelemek által az üzemviteli központnak küldött üzeneteket osztályozza; c) hálózatelemek helyi felügyelete.

A két felső szintet Spoke-rendszer alatt valósították meg (ami a Lisp egy szűkített megvalósítása), mert szerkezete jól illeszkedik a hálózat-üzemvitel feladataihoz és az objektumai igen közel vannak az ISO által értelmezettekhez. A helyi felügyelet a hálózatelemek részét képezi, tehát Eiffel nyelven íródott.

A megvalósítás során a hangsúlyt a Bigaroo modell kialakításának szervezésére helyezték; főleg az objektumkezelés SMF-beli (üzemfelügyeleti rendszerszolgáltatások) feladatait valósítja meg, de tartalmazza az eseménykezelés és a naplózás SMF szolgáltatásait is. A következő szolgáltatásokat lehet igénybe venni: a modellobjektumok létrehozása, módosítása, megjelenítése; az objektum jellemző értékének megváltozását jelző üzenet kezelése; az üzenetek osztályozása; az események naplózása és számlálása. A műveletekre vonatkozó koherenciaszabályok betartását a Spoke-rendszerhez tartozó Héloise-implikátorok biztosítják; ezek a szabályok dinamikusan átalakíthatók. Bigaroo felügyeleti rendszerében 40 osztályt és 52 szabályt írtunk le.

A minta érdekes tanulságokkal szolgált, különösen az objektum fogalmának használatát és az üzemviteli eljárások szerkezetét illetően. Segítségével többek között:
- meg lehetett határozni azt a legkevesebb üzemviteli eljárást, amelyeket a hálózatelemeknek — felügyelhetőségüket biztosítandó — tartalmazniuk kell;
- azonosítani lehetett a CMIS szolgáltatások által megkövetelt összetett szolgáltatások elemeit megvalósító kezelési síkot (pl. osztályozók, objektumok csoportjának kezelése);
- jobban meg lehetett érteni az objektum fogalma alkalmazásának vonzatait a modularitás és a fejlődőképesség területén (az örökösödés alkalmazása);
- mérni lehetett, hogy mennyire alkalmasak az objektumok a hálózat üzemviteli rendszerének modellezésére és megvalósítására;
- mérni lehetett azt, hogy mennyire alkalmasak az objektumorientált elképzelések és eszközök a prototípus (modell) megvalósítására;
- meg lehetett találni az objektumok jelenlegi programozásának néhány hiányosságát (az elképzelések nem szabványosítottak , hatékonyság, elosztottság...).

5.2. Margot [4]

A CNET párizsi A központjának SLC/LCM osztályán megvalósítás alatt áll a távközlési berendezések meghibásodását jelző rendszer modellje (Margot). A tanulmány célja egyrészt az RGT elképzelés gyakorlati megvalósíthatóságának vizsgálata (Q3 felület, modellezés objektumok segítségével), másrészt az elképzelés gyakorlati megvalósítását lehetővé tévő eszközök, különösen az objektumok adatbázisának és az OSI protokolloszlopának vizsgálata.

A modell két SUN munkaállomásból épül fel. Az egyik egy távközlési berendezés, illetőleg közvetítő lényegi egység szerepét, míg a másik a felügyeleti rendszer szerepét játssza. A munkaállomások OSI szerint vannak összekapcsolva, valamennyi kapcsolati réteget megvalósítva. Az alkalmazási rétegben az ACSE, ROSE és CMISE szolgáltatások elemeit alkalmazzuk.

Mindegyik munkaállomás tartalmazza az objektumok adatbázisát. A választás Ontos-ra, az amerikai Ontologic cég termékére esett. Ontos a C++ programozási nyelvhez kapcsolódik.

A felügyeleti rendszer XWindow/Open-Look alatt futó ember–gép felülettel bír, amely a hálózatfelügyelő munkáját teszi lehetővé. A szimulátor rendelkezik egy másik ember–gép felülettel is, amelyen a riasztási események és a riasztási események láncolatából kialakított forgatókönyvek adhatók meg. A forgatókönyveket megadott sorrendben vagy véletlenszerűen játszhatjuk le.

A megvalósított modell főleg a hálózatfelügyelet következő három szolgáltatására támaszkodik: az eseménykezelés, a riasztások kezelése és a naplózás. A felsorolt szolgáltatások maguk pedig az objektumkezelést használják. A szimulátor által keltett rendezett riasztási események a felügyeleti rendszerbe kerülnek, ahol a napló feljegyzi őket, majd a felügyelő előtti képernyőn megjelennek. A felügyelő az alábbi lehetőségekkel bír: megjelenítheti valós időben a (kívánságra osztályozott) riasztásokat; kezelheti és rendezheti a naplókat (létrehozás, törlés, módosítás, keresés, aktiválás); megkereshet egy, a naplóban feljegyzett eseményt.

Számoltunk az objektumok kezelésével kapcsolatos helyi eseményekkel is. Egy objektum létrehozása, egy jel-

lemző érték megváltoztatása például különleges eseményt jelent, amelyet ugyanolyan módon lehet rendezni és feljegyezni, mint a berendezések felől érkező riasztásokat.

A berendezés kialakítása időben változatlan, néhány áramkör és berendezés típusú objektumra korlátozódik, hiszen a minta fő célja a riasztások felügyeletének objektumorientált környezetben, Q3 felületen történő megvalósítása volt.

A kísérlet eredménye lehetővé teszi, hogy megtaláljuk a valamennyi, új technológiát alkalmazó megvalósításnál felmerülő kulcskérdésre adandó válasz néhány elemét: teljesítmény, meghibásodási valószínűség és továbbfejleszthetőség. Ha a választott eszközök beváltják a hozzájuk fűzött reményeket, akkor a minta új hálózatfelügyeleti fejlesztések és kísérletek alapjául szolgálhat.

6. ÖSSZEFOGLALÁS

Az ismertetett gondolatsor az üzemviteli kérdéskör megoldási módjának fejlődését írja le, nem pedig forradalmi változást. A kezelendő kérdések növekvő bonyolultsága, a szabványosítás jelenlegi állása olyan eszközöket és módszereket ad a kezünkbe, amelyekkel jobban csoportosíthatjuk a kérdéseket, kézben tarthatjuk a hálózati üzemvitelre vonatkozó elgondolások fejlődését.

Megemlítjük, hogy az informatika és a távközlés világa igen markánsan közeledik egymáshoz, s a hálózati üzemvitelt egyre inkább a vállalat informatikai vezetési rendszere részének tekintik. Sőt, anélkül hogy túl messzire néznénk a jövőben, világosan látható, hogy a hálózatok üzemvitelének bonyolultsága rohamosan növekszik. A szabályozási, szervezeti, gazdasági és műszaki környezet mind bonyolultabbá válik: egyrészt azért, mert az érintettek (üzemeltetők, hálózatok, ügyfelek) száma növekszik, az együttműködés és az összehangolás kérdései mind élesebben vetődnek fel; másrészt azért, mert folyamatosan növekszik az átviteli kapacitás, ami maga után vonja az üzemviteli adatok mennyiségének növekedését; végül pedig azért, mert az intelligencia egyre jobban szétszóródik a hálózatban, és ezt az elosztott inteligenciát kell felügyelni.

Amennyiben a hálózati üzemvitel eszközei és a szabványok megjelenésével az üzemvitel rendszerezhetővé és fejleszthetővé válik, akkor a hálózat és környezete igen gyorsan fog fejlődni. Az erőfeszítéseket tehát tovább kell folytatnunk, még sok tennivaló van előttünk... A hálózat-üzemvitel lesz-e vajon a modern kor Sziszifusza?

IRODALOM

[1] Murill, B., *Journés Networks'90*, Birmingham, juin 1990.

[2] Arnout, C., Brossard, M., Louvet, O., Risser, N., "An application of the object oriented paradigme to the modelisation of telecommunications services", *Tools 91*, Paris.

[3] Cornilly, J.M., Gérard, M., Gebraud, C., "Expérimentation des concepts d'administration OSI sur la maquette Bigaroo", *Note Technique CNET*, NT/LAA/SLC/.346, 22 mars 1991.

[4] July, P., "Etude et réalisation d'une maguette de système de gestion des fautes d'un équipement de télécommunication", *Note technique CNET*, NT/PAA/OGE/REC/26707, mai 1990.

[5] CCIT, *Recommandations du CCITT*: M.3000 à M.3010.

[6] ISO, Normes no. IS 7498 OSI reference model, IS 9595 CMIS, IS 9596 CMIP, IS 10040 Systems management overview, série 10164 Systems management, série 10165 Structure of management information.

[7] Bounemra, K., "Normalisation des réseaux", *Eyrolles*, 1990.

[8] Pitteloud, J., "Gestion des réseaux selon OSI", *Bulletin technique PTT*, Berne, 2 février 1990.

[9] Sahin, V., Omidyar, C., Bauman, T., "TMN architecture and interworking designs", *IEEE J o9f select areas in communic*, mai 1988, vol.6, no.4.

[10] Boyd, R., Johnson, A., "Network operations and management in a multi-vendor environment", *IEEE communications magazine*, juil 1987. vol.25, no.4.

[11] Okada, H., Yoshida, M., "Toward creation of total operations systems", *NTT review*, jan 1990. vol.2, no.1.

[12] Effinger, M., Henderson, M., "Handing off control to the customer", *Bellcore Exchange*, sep./oct. 1989.

[13] Yamaguchi, H., Yoshida, M., Kobayashi, M., "Customer control of network management from the service provider's perspective", *IEEE communic. mag.*, mars 1990. pp.35–40.

[14] Pangas, B., Otero, M., Peeters, F., "TMN pilot application functions", *XIII International switching symposium*, Stockholm, 27 mai–1er juin 1990.

[15] Bell, J., "How standard can network management be?", *Telecommun. Internat. Ed.*, avril 1990. vol.24. no.4.

[16] Kung, R., Portejoie, J.F., Ségot, H., Stéphant, T., *Perspectives, Commun. et transm.*, avril 1991.

NETWORK MANAGEMENT: STATE OF THE STANDARDS

H. SÉGOT — CNET/PAA/RGE, 38-40, RUE DU GÉNÉRAL-LECLERC F-92131 ISSY-LES-MULINEAUX

C. LECERF — CNET/LAA/SLC, ROUTE DE TRÉGASTEL, BP 40 F-22301 LANNION CEDEX

P. JOLY — CNET/PAA/RGE, 38-40, RUE DU GÉNÉRAL-LECLERC F-92131 ISSY-LES-MULINEAUX

The need for multi-media communication is currently rapidly increasing. This leads to a sharp increase in the number of public and private networks, and they must inter-operate in spite of their being heterogeneous. Mastering those networks is a complex but crucial task. This can only be achieved via good management. Thus, users have been impulsing a heavy thrust in the field of network mamagement and especially its standardization. This paper focusses on the current status of network management standards in the main international bodies, highlighting the main points. It also describes some of France Telecom's experiments concerning the implementation of the standards.

Hubert Ségot az ENST-en szerezte mérnöki diplomáját. 1988 óta dolgozik a hálózatok üzemvitelének területén, a Nemzeti Távközlési Kísérleti Központ (CNET/PAA/RGE) munkatársaként az új eszközök kiválasztásában segíti a France Télécomot. Részt vesz a CCITT szabványosítási munkájában.

Chirstian Lecerf távközlési szakon végzett a DEA-n. Korábban az elosztott vezérlőjű telefonközpontokat tanulmányozta, 1982 óta a többcélú és szélessávú hálózatok üzemvitele és felügyelete területén folytat tanulmányokat a CNET/PAA/RGE-ben. Részt vesz az ISO és a CCITT szabványosítási munkájában.

Pascale Joly a párizsi Central-on és az ENST-n végzett mérnök, 1990 óta tanulmányozza a távközlési hálózatok üzemvitelét, különös tekintettel a kapcsolástechnikára. A CNET/PAA/RGE-n belül megvalósíthatósági kísérleteket végez. Részt vesz az ETSI szabványosítási munkájában.

PROTOKOLLOK SPECIFIKÁLÁSA ÉS TESZTELÉSE

TARNAY KATALIN

KFKI MÉRÉS- ÉS SZÁMÍTÁSTECHNIKAI KUTATÓ INTÉZETE
BUDAPEST, PF. 49. H-1525

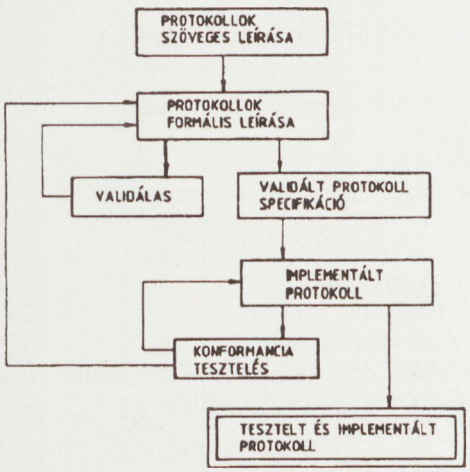
A távközlési szoftverek egyik fontos alapelemét, a protokollt mutatja be a cikk. A protokollok matematikai modelljei közül a véges automatákra épülő modelleket hangsúlyozzuk, amely a legtöbb specifikálás alapja.

1. BEVEZETÉS

A kommunikációs protokollok a távközlési szoftverek egyik fontos csoportját képezik. A protokoll — első közelítésben — azoknak a szintaktikus, szemantikus és időbeli szabályoknak az együttese, amelyek a kommunikációs szolgáltatásokat biztosítják. A távközlési szoftver el van osztva a hálózat közeli és távoli csomópontjaiban. A csomópontokban lévő beépített protokollelemek és processzek koordinálják és szinkronizálják az elosztott hálózati elemek közti együttműködést.

Bár a legtöbb protokollt a nemzetközi szabványügyi szervek (CCITT, ISO) vagy nagy számítógépgyártó cégek (IBM, DEC, stb.) részletesen specifikálják, mégis számos probléma merül fel megvalósításuk és ellenőrzésük során. A szöveges leírások nem egyértelműek és nem teljesek. A megoldás a formális specifikálás. A formális leíró technika (Formal Description Technique, FDT) egy rendszer jellemzésének matematikailag megalapozott módszere, amely egyértelmű absztrakt modellre épít. A formális nyelvi eszközökkel specifikált rendszer megvalósítását és ellenőrzését a korszerű szoftvertechnológia eszköztára is támogatja [8].

Egy protokoll életciklusát a következőképpen foglalhatjuk össze. A protokollokkal szemben támasztott követelmények szöveges megfogalmazását a formális specifikálás követi, amely validálható. A validált protokollspecifikáció már lefordítható valamilyen futtatható nyelvre és a megvalósítás, az implementálás tesztelhető vagyis összehasonlítható a formális specifikációval. A protokoll életciklusával és az életciklus egyes lépéseinek végrehajtását segítő eszközökkel egy új szakterület, a protokolltechnológia (protocol engineering) foglalkozik (1. ábra).



1. ábra. A protokoll technológia lépései

A protokolltechnológia tárgya a kommunikációs protokollok modellezése, formális specifikálása, validálása, implementálása és konformancia tesztelése. A protokolltechnológia egyik ága a számítógéppel segített protokolltechnológia (Computer-Aided Protocol Engineering, CAPE), amely a protokolltechnológia színvonalas szoftvereszközeit tartalmazza.

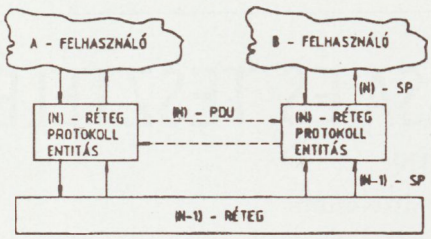
A cikk célja, hogy áttekintést adjon a protokollok specifikálásáról és teszteléséről. A cikk 5 fejezetből áll. A bevezetést követő 2. fejezet ismerteti a főbb protokollmodelleket: a kommunikáló véges automatát, formális nyelvtanon alapuló modellt, gráfokat és algebrai modelleket. A 3. fejezet a protokolltechnológiával foglalkozik, bemutatja a specifikálás, validálás, implementálás és konformancia tesztelés alapproblémáit. A 4. fejezetben röviden ismertetünk két, véges automata modellre épülő specifikációs nyelvet, az SDL-t (Specification and Description Language) és Estelle-t (Extended State–Transition Language), valamint az algebrai specifikáció nyelvét, a LOTOS-t (Language of Temporal Ordering Specification) és annak két összetevőjét, a CCS-t (Calculus of Communicating Systems) és az ACT ONE-t. Az 5. fejezet a protokolltesztelés alapfogalmait, a tesztelési előírásokat és a nemzetközi megállapodásoknak megfelelő tesztelési előírások jelölésmódját, a TTCN-t (Tree and Tabular Combined Notation) és alkalmazását tárgyalja.

2. PROTOKOLL MODELLEK

2.1. A protokoll

A protokoll két kommunikáló entitás közti adatforgalmat határozza meg, előírja a kooperáció szintaktikus, szemantikus és időbeli szabályait. Elképzelhető lenne, hogy egy hálózatban egyetlen protokoll legyen, ami egyetlen nagy és összetett szabálygyűjtemény. Ez a gyakorlatban annyira bonyolult, hogy megvalósítása és üzemeltetése szinte lehetetlen. Ezért a szabályokat feladattípusok szerint csoportosították és így rétegezett rendszert alakítottak ki, amelynek réteghatárai pontosan definiáltak. A feladattípusok, funkciók rétegenként jól elkülönülnek. Egy-egy rétegben a funkciókat a protokollentitás valósítja meg. Az azonos rétegekben levő protokollentitások — ezek gyakran processzek — három határfelületen (2. ábra) keresztül cserélnek információkat: a felső és alsó réteghatáron a szolgálati primitívek (Service Primitive, SP) haladnak át, a szomszédos, azonos rétegbeli protokollentitások pedig protokoll adatelemeket (Protocol Data Unit, PDU) váltanak. A protokoll definíciója leginkább egy nyelv definíciójához hasonlít. A következőket kell definiálni:

- az érvényes üzenetek pontos formátumát (szintaxis),
- az információcsere eljárásának szabályait (nyelvtan),



2. ábra. A protokoll entitások információcseréje

- végül az érvényes üzenetek szótárát jelentésükkel együtt (szemantika).

A protokollok és a beszélt nyelv közti hasonlóság sok protokollprobléma megoldásában segíthet.

Protokolltulajdonságok

A protokollnak általános és speciális tulajdonságai vannak. Az általános tulajdonságok minden protokollra jellemzőek, a speciális tulajdonságok csak egyes protokollra vagy protokollcsoportokra. Az általános protokolltulajdonságok közül a legfontosabbak a következők:

- holtpont-mentesség,
- végállapotok garantált elérése, vagyis terminálás,
- teljesség, amely valamennyi lehetséges előforduló helyzet megoldására rendelkezéseket tartalmaz,
- ciklusok hiánya,
- stabilitás, vagyis annak biztosítása, hogy a protokoll rendellenes működés után visszatér normál állapotába.

Speciális tulajdonságokra példa a megjelenítési rétegprotokollok kódkonverziója vagy a virtuális terminálprotokollokban a valós terminálokra vonatkozó előírások.

A továbbiakban még érdemes az általános protokolltulajdonságokhoz kapcsolódó protokollhibákat felsorolni:

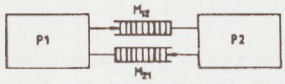
- holtpont: olyan stabil globális állapot, amelyből nem lehet üzenetet küldeni, vagyis a két kommunikáló processz két vevőcsatornája között üres a csatorna;
- specifikálatlan vétel: nincs definiálva a következő lépést meghatározó következő állapotfüggvény;
- nem végrehajtható kölcsönhatás: olyan adás vagy vétel, ami specifikált, de nem végrehajtható.

Egy kölcsönhatás akkor és csakis akkor végrehajtható, ha a globális állapot elérhető:

- túlfolyás: a csatorna nem korlátos; a protokoll csak akkor korlátos csatornájú, ha minden csatornája korlátos;
- állapot-ellentmondás: az egyik processznek van olyan állapota, amely a másik processz két vagy annál több állapotával egyidejűleg is stabil módon fennállhat;
- jól formáltság hiánya: a jól formáltság azt jelenti, hogy az üzenet vétele akkor és csakis akkor specifikált, ha végrehajtható, vagyis nincsenek benne nem specifikált vagy nem végrehajtható üzenetvételek, ennek hiánya protokollhiba;
- életképesség hiánya: az életképesség azt jelenti, hogy minden elérhető állapotból elérhető minden állapot. Ha a protokoll ciklikus (kezdő és végállapota azonos), akkor a következő feltétel is elegendő: minden globális állapot elérhető a kezdeti állapotból, ennek hiánya protokollhiba;
- terminálás hiánya: a kezdeti állapotból indítva, a protokoll nem ér el mindig egy végállapotba.

2.2. Kommunikáló véges automaták

A protokollok alapvető modellje [3] kommunikáló véges automatákra épül (3. ábra). A véges automaták a folyamatokat, a várakozó sorok a csatornákat reprezentálják. Minden egyes folyamatnak egy véges automata felel meg. Minden egyes kommunikáló folyamatpárt egy csatorna köt össze, amely hibamentes és FIFO-elv szerint működik.



3. ábra. Kommunikáló véges automaták

A protokoll egy rendezett négyes

$$P = (Q, o, M, succ),$$

ahol
Q az állapotok halmaza,
o a kezdeti állapotok halmaza,
M az üzenetek halmaza,
$succ$ a leképező függvény.

Ha a protokoll két folyamatot tartalmaz, akkor

$$Q = (Q_1,\ Q_2),$$
$$o = (o_1,\ o_2),$$
$$M = (M_{12},\ M_{21}),$$
$$succ = (succ_1,\ succ_2),$$

ahol
Q_1 az első, Q_2 a második folyamat állapotainak halmaza,
o_1 az első, o_2 a második folyamat kezdő állapotainak halmaza,
M_{12} az első folyamatból a második, ill.
M_{21}a második folyamatból az első folyamatba irányuló csatornán áthaladó üzenetek halmaza,
$succ_1\ (s_1, x)$ az első folyamat leképező függvénye és s_1 azt az állapotot jelenti, amelybe az első folyamat kerül x érkezése után ($s_1 \epsilon Q_1$ és $x \epsilon M_{21}$), ill.
x küldése után ($s_1 \epsilon Q_1$ és $x \epsilon M_{12}$).
$succ_2\ (s_2, x)$ a második folyamat leképező függvénye, értelmezése $succ_1(s_1, x)$-hez hasonló.

E modellben a folyamatok véges automaták, amelyeket a következő rendezett ötös jellemez:

$$FSM = (X, Z, Q, f_z, f_s),$$

ahol
X a bemenő ábécé elemeinek halmaza,

$$X = (x_1, \ldots x_p),$$

Z a kimenő ábécé elemeinek halmaza,

$$Z = (z_l \ldots z_q),$$

Q az állapotok halmaza,
f_z a kimenő függvény,
f_s a „következő állapot" függvény.

A kimenő függvényt és a „következő állapot" függvényt a következő összefüggések definiálják

$$z_v = f_z(x_v, s_v),$$
$$s_v + 1 = f_s(x_v, s_v),$$

ahol v az állapotátmenet aktuális száma.

Az alapmodellben feltételeztük, hogy két véges automata FIFO-csatornákon keresztül kommunikál. Az alapmodellt azért kell módosítanunk, mert a protokollspecifikációban szereplő folyamatok a gyakorlatban nem véges automaták és a folyamatok nem FIFO-csatornákon keresztül kommunikálnak. Jobb közelítést kapunk, ha a véges automatákat absztrakt gépekkel helyettesítjük és a FIFO-csatornákat szemaforokkal vezéreljük.

Miért jellemzik jobban az absztrakt gépek a valóságos protokollokat?

A valóságos protokollok működését állapotátmenet-táblákkal szokás leírni. Ezek az állapotátmenet-táblák eltérnek a véges automaták állapotátmenet-tábláitól, mert predikátumokat és akciókat is tartalmaznak. A predikátumok igaz vagy hamis értékétől függően a következő állapot, valamint a kimenet is eltérő. Az akciók olyan események, amelyekre nem állapotváltozás és új kimenő üzenet jellemző, hanem valamilyen cselekménysorozat (pl. óra indítása).

A valóságos protokollok bemenő és kimenő üzenetei változókat is tartalmaznak. A protokoll működését a változók értékei (pl. ablakméret) is befolyásolják.

A protokollt leíró gép [5] egy rendezett hetes

$$ASM = (X, Z, Q, VAR, PRED, ACT, TRANS),$$

ahol

X, Z, Q értelmezése azonos a véges állapotú gép hasonló nevű halmazainak definíciójával,

VAR a változók halmaza, melyre igaz, hogy

$$VAR = VARL\ VARG$$

ahol

$VARL$ a lokális változók halmaza,

$VARG$ a globális változók halmaza,

$PRED$ a predikátumok halmaza,

ACT az akciók halmaza, minden akció két diszkjunt halmazra bomlik:

$$ACT = ACTL \cup ACTG$$

$$ACTL \cap ACTG = 0$$

ahol

$ACTL$ csak lokális változóktól függ,

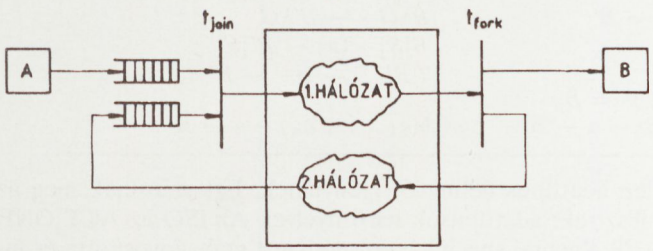
$ACTG$ globális változóktól függ,

$TRANS$ az átmenetek halmaza.

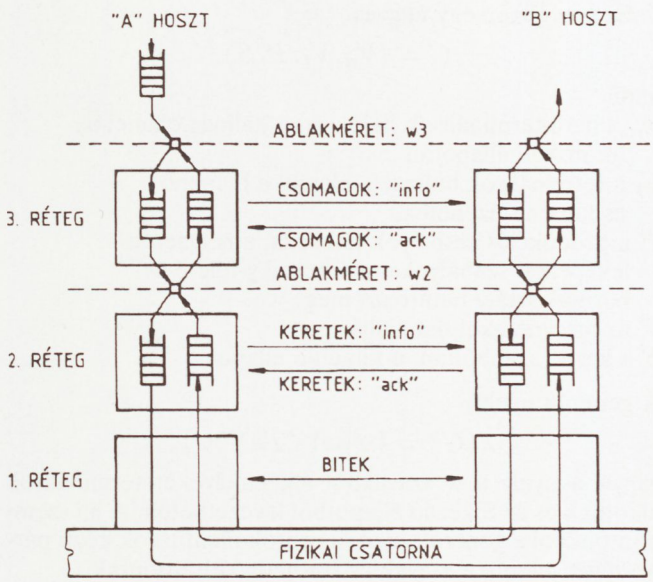
Minden $t \in TRANS$ a következő alakú:

$$t = (s_1, i, p, act, s_2, o),$$

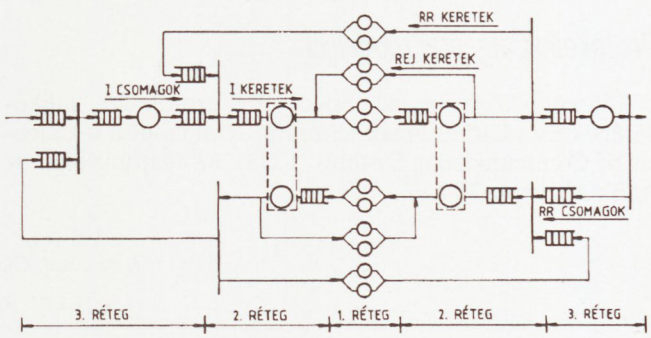
ami azt jelenti, hogy ha az absztrakt gép s_1 állapotában i bemenő üzenetet kap és a predikátum igazság értéke p, akkor a gép s_2 állapotba megy át, o kimenő üzenetet küld és act nevű akciót kezdeményez.



4. ábra. Szemaforos protokollmodell



5a. ábra. X.25 protokollok



5b. ábra. X.25 protokollok szemaforos modellje

A másik kérdés, miért adnak jobb közelítést a szemaforos protokollmodellek, mint a FIFO-sorok. A FIFO-sor visszacsatolás nélküli, egyirányú kommunikációt, monológot tételez fel, holott a protokollentitások közti üzenetváltás mindig dialógus. A szemaforvezérlés sokkal jobban tükrözi (4. ábra), hogy a kapcsolat dialógus jellegű és nem két párhuzamos, ellenkező irányú monológ [11]. Az ábrán csak A küld üzeneteket B-nek és akkor és csakis akkor, ha a t_{join}-on keresztüli visszacsatolás engedélyezi és a szemafor nyit.

A szemaforok másik előnye, hogy nem kell a protokollentitást környezetéből kiragadni, hanem a szomszédos rétegentitások is kapcsolódhatnak. Az 5. ábra szemléletesen mutatja az X.25 három rétegének szemaforos protokollmodelljét. Az egyszerűség kedvéért itt is feltételeztük, hogy az információforgalom egyirányú. Az ábrán jól követhető, hogy egyaránt meg kell érkeznie a keretek és a csomagok nyugtázásának is, tehát a szomszédos rétegek protokollentitásainak helyesen kell működnie.

Reguláris nyelvtannal jellemzett protokollmodellek

A 2. fejezet bevezető részében a protokollt a nyelvhez hasonlítottuk. A nyelv párbeszédekből, a párbeszéd mondatokból áll. E mondatokat reguláris nyelvtanokkal generálhatjuk [14], [15].

Ez a nyelvtan egy négyes

$$G = (V_n, V_t, P, S),$$

ahol

V_n a nem-terminálisok halmaza, a halmaz elemei a protokoll állapotai,

V_t a terminálisok halmaza, elemei a bemenő és kimenő üzenetek,

P a produkciós szabályok halmaza, amelyben a leképezési szabályokat a Chomsky-féle 3-as normálosztály határozza meg

P az átmeneteket definiálja,

S a kezdő szimbólum, a kiinduló állapot $S\ V_n$.

A generált nyelv

$$L(G) = \{w | wV^*TsS^*w\},$$

vagyis a nyelv a w szavakból áll, amelyeket terminálisok alkotnak és az S kezdő állapotból levezethetők. A mi szempontunkból a generált nyelv a protokollentitások közti párbeszédet jelenti, a szavak az üzenetekkel azonosak.

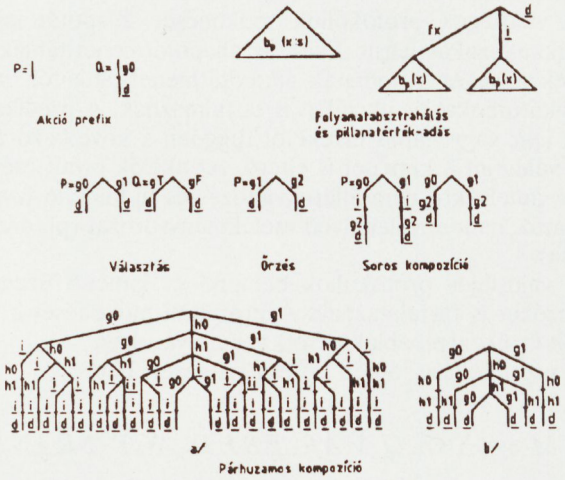
A formális nyelvtannal való protokollmodellezés előnye, hogy az alapmodell komponens nyelvtanokkal egyszerűen bővíthető.

Protokollok algebrai modelljei

Két különböző modell terjedt el a protokollok működésének és adattípusainak leírására. A működést a Calculus of Communicating Systems (CCS), az adattípusokat az ACT ONE specifikálja.

A CCS osztott rendszerek modellezésére készült. Milner [19] ötlete volt, hogy a rendszereket a környezetükkel való kölcsönhatásokkal határozza meg. A kölcsönhatásokat operátorokkal írja le (1. táblázat), amelyek szemléletes jelentése a 6. ábrán látható.

Az ISO egyik szabványos nyelvének, a LOTOS-nak [4] a CCS az egyik összetevője. A CCS-t, kicsit leegyszerűsítve, a kommunikáció algebrájának tekintjük az adattípusok definiálása nélkül.



6. ábra. CCS viselkedési fák

1. táblázat. CCS kölcsönhatások

CCS* SZINTAXIS

Operátor	B	Megjegyzés
Összegzés	ΣB	+ jelet is használnak
Akció prefix	$g!e.B1$	e érték kifejezés
	$i.B$	
Kompozíció	$B1 \mid B2$	
Tiltás	$B \backslash G$	G a kapuk halmaza
Újracímkézés	$B1[S]$	az újracímkézés
Őrzés	$e \rightarrow B1$	e Boole-típusú
Viselkedés pillanatértékek	$b(e_1, \ldots, e_n)$	b azonosító

CCS* SZEMANTIKA

Operátor	Feltétel	Követelmény
Összegzés	$B \in B,\ B - a \rightarrow B$	$\Sigma B - a \rightarrow B'$
Akció prefix	nincs	$g!e.B - ge \rightarrow B$
		$i.B - i \rightarrow B$
Kompozíció	$B1 - a \rightarrow B1'$	$B1 \mid B2 - \rightarrow B1' \mid B2$
	$B2 - a \rightarrow B2'$	$B1 \mid B2 - a \rightarrow B1 \mid B2'$
	$B1 - a \rightarrow B1',\ B2 - a \rightarrow B2'$	$B1 \mid B2 - i \rightarrow B1' \mid B2'$
Tiltás	$a = gv, g \notin G, B - a \rightarrow B'$	$B \backslash G - \rightarrow B' \backslash G$
Újracímkézés	$B - \rightarrow B'$	$B[S] - (a) \rightarrow B'[S]$
Őrzés	$B - a \rightarrow B'$	$TRUE \rightarrow B - \rightarrow B'$
Viselkedés pillanatértékek	$b(x_1 : s_1, \ldots, x_n : s_n) := B_b$	
	$[e_1/x_1, \ldots, e_n/x_n]B_b - a \rightarrow B'$	$b(e_1, \ldots, e_n) - a \rightarrow B'$

A protokollok formális specifikációja nem lehet teljes az absztrakt adattípusok matematikai modellje nélkül. A programozási nyelvek implementációfüggő konkrét adattípusokat definiálnak, a protokollok formális leíró technikái azonban adatstruktúra-, implementálás- és kódfüggetlen adattípus definiciót igényelnek. Ezt valósítják meg az absztrakt adattípusok leíró nyelvei. Az ISO az ACT ONE [10] algebrai specifikációs módszert szabványosította és integrálta a LOTOS-ba.

Egy absztrakt adattípus specifikációja egy hármas:

$$SPEC = (S, OP, E)$$

ahol
S a fajták (sorts) halmaza,
OP a műveleti szimbólumok halmaza,
E az egyenlőségek halmaza.

Az adattípus-specifikációk atomiak vagy komplexek. Az atomi specifikáció:
- egyszerű,
- formális,
- parametrizált.

A komplex specifikáció:
- kombináció,
- modularizálás,
- újraelnevezés,
- aktualizálás.

3. PROTOKOLLTECHNOLÓGIA

3.1. A protokolltechnológia elemei

A protokolltechnológia (protokoll engineering) tárgya a protokollok tervezése, megvalósítása és ellenőrzése. A tervezés első lépése a felhasználói követelményekből formális specifikáció előállítása.

A protokollspecifikálás a protokoll definíciójának formális modellel való meghatározását jelenti. Ez a formális modell a protokollok szöveges leírását fogalmazza meg egyértelműen és algoritmizálható módon. A protokollmodell logikai helyességét (pl. holtpont-mentesség) ellenőrizni kell a megvalósítás előtt. Különböző validáló eljárások terjedtek el a logikai helyesség ellenőrzésére. A validálás eredményétől függően vagy módosítani kell a protokollmodellt, esetleg a felhasználói követelményeket is, vagy sor kerülhet a protokoll megvalósítására. Az implementálás a protokollspecifikáció megvalósítása futtatható formában a környezeti feltételek figyelembevételével. Az implementált protokollt is ellenőrizni kell, hogy megfelel-e a protokollspecifikációnak. Ez az ellenőrzés a konformancia tesztelés vagy a ma egyre inkább terjedő magyar szóhasználattal: alkalmasságvizsgálat [24], [25].

A protokolltechnológia további fontosabb elemei a protokollkonverzió, a protokollszimuláció és teljesítmény-előrejelzés. A protokolltechnológia eszköztára a specifikáló nyelvektől (SDL, ESTELLE, LOTOS stb.) a protokollfejlesztő munkaállomásokig (pl. PROCONSUL, PROTEAN) terjed.

Protokoll specifikálása

Már a főbb tulajdonságok felsorolása is arra utal, hogy helyesen működő kommunikációs protokollt informális leírás alapján nem lehet tervezni. Bármely protokollmegvalósításban sok hiba, valamint a váratlan és nem kívánatos jelenségek egész sorozata léphet fel. Ezért fontos, hogy a protokolloknak matematikailag egzakt leírása legyen.

Az osztott rendszerek kommunikációs architektúrája különböző protokollrétegek architektúrájának tekinthető. Minden egyes réteg a felhasználójának szolgálatokat nyújt. A szolgálatok specifikálása a protokollok bemenő és kimenő viselkedését határozza meg. A szolgálatspecifikáció absztrakt olyan értelemben, hogy leírja a parancsok típusát és hatásukat, de nem foglalkozik a parancsok formátumával és a szállításuk mechanizmusával (pl. rendszerhívások, interruptok). Általában a szolgálatspecifikáció azon szolgálati primitívek halmaza, amelyek a réteg interfész-illesztőn keresztül elérhető működési eljárásokat szabályozzák.

A felhasználó a szolgálati primitíveket „érzékeli", de a protokolltervezőnek a protokoll belső struktúráját kell megterveznie. A protokollspecifikáció meghatározza a protokollentitás működését:
- hogyan válaszol a felhasználó parancsaira,
- hogyan válaszol a szolgálatnyújtónak,
- hogyan kezeli a belső kezdeményezésű tevékenységeket.

A protokollspecifikálás első közelítésben a protokollmodell matematikailag pontos és hű leírását jelenti. E modellek felhasználásával különböző, kommunikációs célú specifikációs nyelveket dolgoztak ki:
- SDL (kiterjesztett FSM),
- ESTELLE (kiterjesztett FSM),
- LOTTOS (CCS + absztrakt adattípus),
- ASN.1 (adatstruktúrák + APDU kódolása).

A protokollspecifikálás első lépése a felhasználói követelmények megfogalmazása. Ha teljes vagy részleges protokolltervezésre van igény, akkor a protokollspecifikálás a protokollszintézis valamelyik módszerével egészül ki. A leggyakoribbak:
- produkciós szabálytechnika [27],
- struktúrált szintézis [6],
- lépésről lépésre történő finomítás technikája [13].

Mindhárom módszerben közös, hogy a feladatot dekomponálja, és egyszerűsített kommunikáló véges automatákból építkezik.

Verifikálás

A verifikálás annak bizonyítása, hogy a protokoll logikailag helyesen és a felhasználói követelményeknek megfelelően működik. A verifikálást a tervezési fázisban, még mindenképp az implementálás előtt kell elvégezni. A 2.1. pontban felsoroltuk azokat a protokolltulajdonságokat, logikai előírásokat, amelyek a protokoll helyes működését meghatározzák [18]. A verifikálási módszerek közül legelterjedtebb az elérhetőségi analízis [3] [13] [17] [20] [23] [26] [27] [28].

Az elérhetőségi analízis során a kommunikáló véges automaták globális állapotterét elérhetőségi gráfként ábrázoljuk és a lehetséges kölcsönhatásokat elemezzük. Fő probléma a kombinatorikus robbanás, ennek megelőzésére különböző ravasz dekomponálási eljárásokat dolgoztak ki.

További megoldást jelent a numerikus Petri-hálóval történő verifikálás, valamely folyamatalgebra alkalmazása (pl. CCS-ben a globális állapottér előállítható az expanziós tétel alkalmazásával), továbbá a szimbolikus végrehajtás.

Egy protokoll algoritmikus verifikálása abban az esetben lehetséges, ha a protokoll leírható véges állapotú térrel. A jól strukturált protokollspecifikáció segít a túlzott komplexitás okozta hátrányok elkerülésében. Érdemes megjegyezni, hogy a legtöbb eljárás feltételezi, hogy az állapotátmenetek determinisztikusak, az átmenetek időtartama zérus és a csatornák FIFO-elv szerint továbbítják az üzeneteket.

Implementálás

A protokollok formális specifikációja az első lépés az automatikus implementálás felé. A formális specifikációból kiindulva háromféle implementálás lehet:
- kézi,
- félautomatikus,
- automatikus.

A kézi implementálás a hagyományos megvalósítást jelenti.

A félautomatikus implementálás során először a keresztfordító a specifikációs nyelvből magas szintű nyelvi kódot generál, pl. ESTELLE-ből C nyelvűt. A félautomatikus implementálás második részében a végrehajtható kódokat állítja elő a rendszer. A környezetfüggő részt csak manuálisan lehet megadni.

Az automatikus implementálás ma még inkább csak cél és törekvés, mert nehéz elképzelni, hogy az operációs rendszerrel és a környezettel való interfész megvalósítása hogyan automatizálható. 70 %-os és azt megközelítő automatizálás ma még maximális célkitűzésnek tűnik.

Konformancia tesztelés

Az OSI célja, hogy heterogén rendszerek, amelyeket egymástól függetlenül és különbözőképpen valósítottak meg, együttműködhessenek. Az együttműködés feltétele a szabványoknak megfelelő protokoll. A konformancia tesztelés, más szóval az alkalmasságvizsgálat követelményrendszerét úgy kell kidolgozni, hogy a megvalósított protokoll és a szabvány azonosságát egyértelműen igazolja. A követelmények tartalmi osztályozása a protokoll következő elemeire vonatkozó előírásokat tartalmazza:

- üzenetek és paramétereik,
- időzítések,
- állapotok,
- protokollmechanizmusok.

A konformancia tesztelés lényege tesztsorozatok generálása és a válaszsorozatok értékelése. A konformancia tesztelésnek két típusa van. Egyik esetben a tesztsorozatokat és válaszsorozatokat a protokoll formális leírásából származtatjuk, a másik esetben szabványos előírások határozzák meg mindkét sorozatot. A szabványos előírásokra külön jelölésmódot dolgoztak ki, a TTCN-et.

Protokollszimulálás

A protokollszimulálásnak kétféle értelmezése van [22]. Az egyik értelmezés szerint a szimulálás azt jelenti, hogy egy referenciaprotokollt valósítunk meg, melynek minden funkcióját és mechanizmusát elméletileg jól megalapozott, gondosan magvalósított módszerekkel ellenőrizzük. A másik értelmezés szerint a protokoll környezetét is szimuláljuk. Ha a specifikációs nyelvet teljesítményprimitívekkel egészítjük ki [1], akkor a rendszer különböző teljesítményvonatkozásai (késés, erőforrások kihasználtsága, sztochasztikus viselkedés) is előre jelezhetők. Eredményes kísérletek történtek protokollok diszkrét eseményszimulációjára GPSS-szel. A GPSS-modell működtetése során a felhasználó különböző intenzitású forgalmat írhat elő, az eredményeket a rendszer automatikusan előállítja és kijelzi.

A referenciaprotokoll jellegű szimulálásokat főleg konformancia teszterekben, a teljesítményjellemzőkkel kiegészített protokollszimulálást a forgalom, az ablak, a nyugtázás, hibák és különböző várakozások kölcsönhatásainak vizsgálatára használják.

4. PROTOKOLLOK SPECIFIKÁCIÓS NYELVEI

4.1. Specifikációs nyelvek

A távközlési szoftvereknek a real-time természete, nagy bonyolultsága és a szükséges nagy megbízhatóság indokolja a korszerű formális nyelvi eszközök, különösen a kommunikációs célú specifikációs nyelvek alkalmazását. Bár több cég és intézmény dolgozott ki specifikációs nyelveket, mégis a legelterjedtebbek a CCITT és ISO nyelvei.

2. táblázat. Kommunikációs célú specifikációs nyelvek összehasonlítása

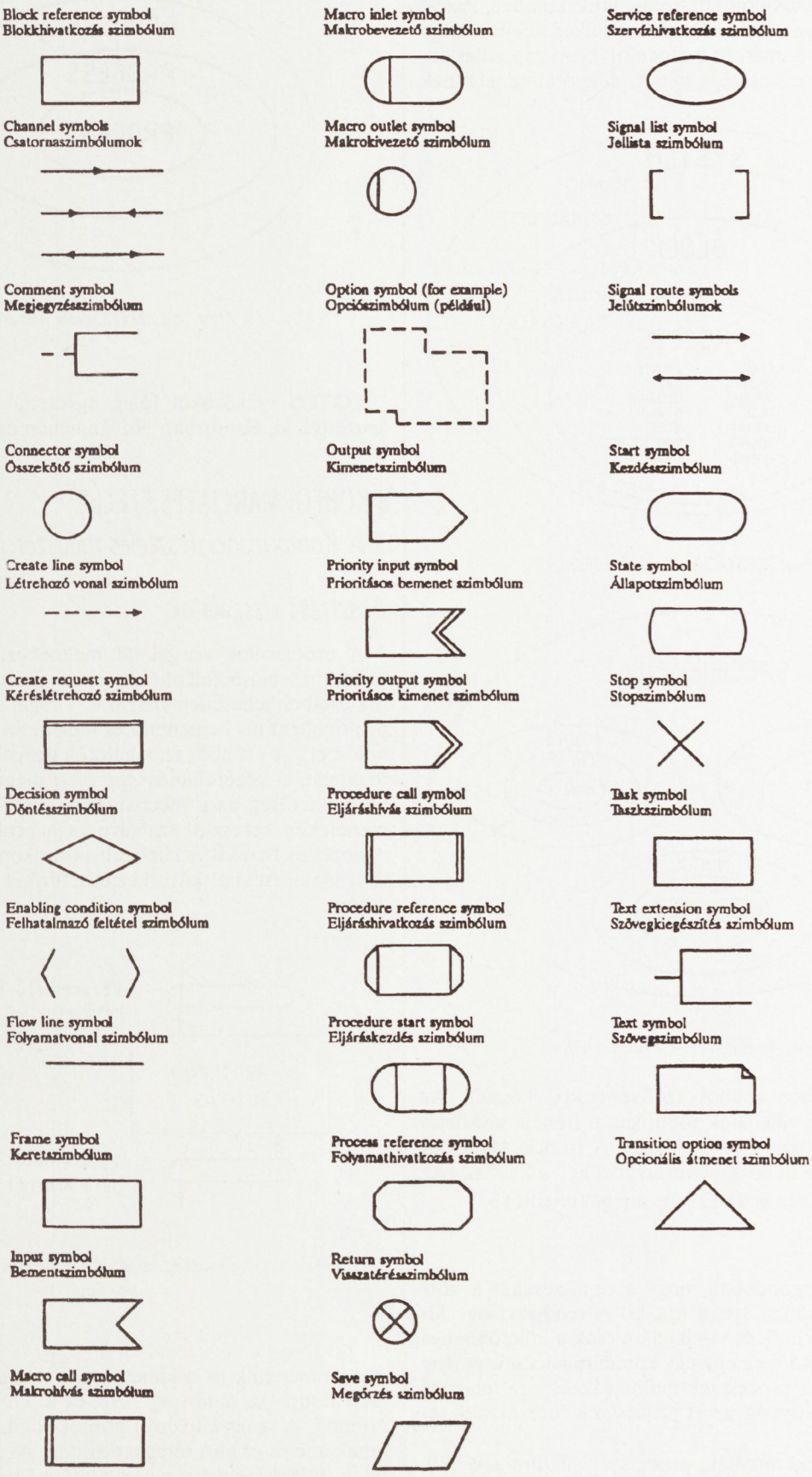
	VÉGREHAJTÁS SORRENDJE	ELEMI MODULOK KÖZTI KOMMUNIKÁCIÓ	ADATTÍPUSOK
SDL	EFSM	kölcsönhatási pontok, várakozó sorok, globális processzazonosító	ADT
ESTELLE	EFSM	kölcsönhatási pontok, várakozó sorok	PASCAL
LOTOS	CCS	rendezvous, kapu	ADT ACT ONE

A CCITT a különböző kommunikációs rendszerek specifikálására az SDL-t (Specification and Description Language) dolgozta ki, melyet a Z.100 sorozat ismertet. Az ISO két specifikációs nyelvet ajánl, elsősorban protokollok specifikálására. Az egyik nyelv az Estelle (Extended state transition language) az SDL-hez hasonló, a másik nyelv a LOTOS (Language for Temporal Ordering Specification) az előző fejezetben említett CCS-re és ACT ONE-ra épül. A három nyelvet a 2. táblázat hasonlítja össze.

4.2. SDL és ESTELLE nyelvek

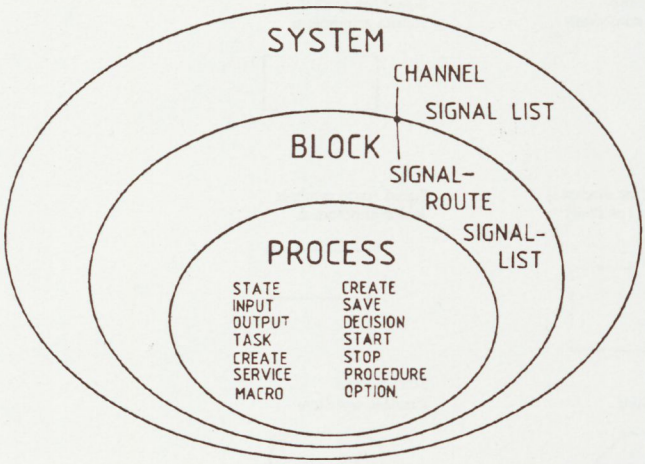
Mindkét nyelv a kiterjesztett véges automata modellre épül és hierarchikus felépítésű. Az SDL szintaxisát grafikusan és programnyelven, az Estelle-t csak programnyelven valósították meg. Az SDL/GR a grafikus szintaxist a grafikus szimbólumok halmazával valósítja meg, amelyek között szabályokat definiál. A szimbólumok és a szabályok egyértelműen definiálják egy rendszer viselkedését. Az SDL/PR szimbólumai kulcsszavak és a szabályai nagyon hasonlóak a PASCAL nyelv szabályaihoz. A 7. ábra szemléltet szimbólumokat.

A nyelvi hierarchia legmagasabb szintje a rendszer, ezt követi a blokk, a blokkot a processz. Az SDL nyelv hierarchiája jól látható a 8. ábrán.

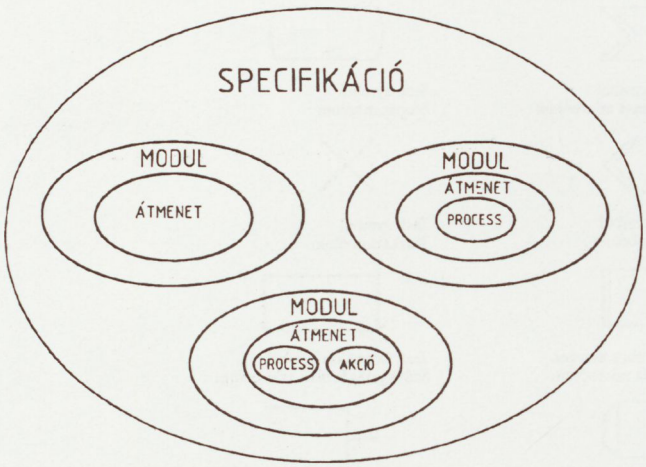


7. ábra Az SDL grafikus szimbólumai

Az Estelle sok szempontból (matematikai modell, Pascal elemek) hasonló az SDL-hez, annak grafikus kifejező ereje nélkül. A hierarchia más, itt a modulok közti kapcsolat dominál és nagyobb jelentősége van az állapotátmeneteknek (9. ábra).



8. ábra. Az SDL nyelv hierarchiája



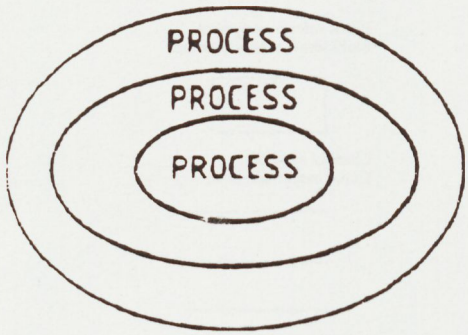
9. ábra. Az Estelle nyelv hierarchiája

Mindkét nyelvhez számos szoftvereszköz készült. Az európai távközlési vállalatok többsége, a francia távközlés kivételével, SDL eszközöket használ. A francia távközlés saját fejlesztésű Estelle szoftvereszközöket alkalmaz. Az Egyesült Államokban is az Estelle a legelterjedtebb.

4.3. LOTOS nyelv

A LOTOS alapgondolata, hogy a rendszereket a környezetükkel kölcsönhatásban írja le. A rendszert egy fekete doboznak tekintik és viselkedése csak a kölcsönhatási pontokon figyelhető meg egy-egy kommunikációs esemény során. A rendszert processzek definiálják és egy rendszer lényegében egymásba ágyazott processzek sorozatának tekinthető.

Így a LOTOS hierarchiája processzek hierarchiája (10. ábra). A processzspecifikálás dinamikus részét viselkedési kifejezések, adattípusait egy speciális absztrakt adattípusdefiniáló nyelv, az ACT ONE írja le. A nyelv ereje, hogy a dinamikus viselkedések jellemzésére speciális operátorokat vezet be.



10. ábra. A LOTOS nyelv hierarchiája

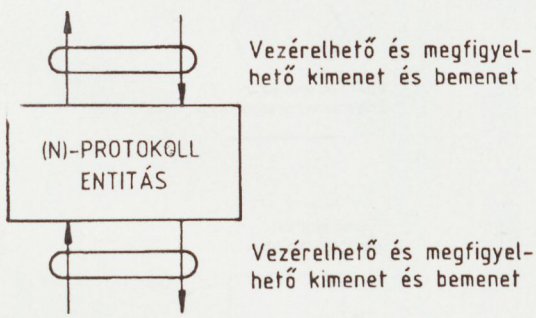
LOTOS eszközöket főleg egyetemi környezetben fejlesztettek ki, elsősorban Hollandiában és Kanadában.

5.KONFORMANCIATESZTELÉS

5.1. Konformanciatesztelés módszerei

Absztrakt vizsgálatok

A protokollok vizsgálatát megnehezíti, hogy a protokollentitás belső felépítését, belső működését nehéz, sőt sok esetben lehetetlen vizsgálni. Vizsgálataink során csak a protokollentitás bemenetei és kimenetei „láthatók" és „érhetők el". A 11. ábra szemlélteti a vizsgálatok két alapvető fogalmát: a vezérelhetőséget és a megfigyelhetőséget. A vezérelhetőség az a mechanizmus, amely az elérhető bemeneteken keresztül szabályozza a protokoll működését. A vezérlés hatását a kimeneti pontokon figyelhetjük meg és a kimenetekből következtethetünk a protokoll viselkedésére.



11. ábra. Az alkalmasságvizsgálatok vezérelhető és megfigyelhető pontjai

A bemenetek és a kimenetek a szolgálatelérési pontok (12. ábra). Az ábrán az ASP és a PDU többesszámban értendő. A szolgálatelérési pontok logikai interfészt jelentenek a felső és alsó réteghatárokon. Az (N)-entitás üzeneteket küld és fogad a szolgálatelérési pontokon keresztül. Ezek az üzenetek az absztrakt szolgálati primitívek (SP) és a protokoll adatelemek (PDU). A szolgálati primitíveket azért nevezzük absztraktnak, mert csak típusokat és paraméterlistájukat specifikáljuk, de sem pontos, valamilyen karakterkóddal jellemzett formátumokat, sem pedig

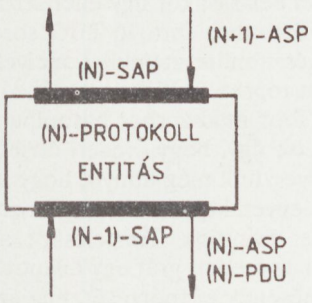
a szükséges rendszerhívásokat, megszakításokat, eljáráshívásokat, amelyek a számítógéprendszerhez tartoznak, nem definiáljuk. A vizsgálati módszert azért nevezzük absztraktnak, mert a megfigyelt és vezérelt protokollelemek absztrakt szolgálati primitívek.

A vizsgálat — erősen leegyszerűsítve — két részből áll: vezérlésből és megfigyelésből. A nemzetközi szabványjavaslatok meghatározzák a vezérlési és megfigyelési pontokat (Points of Control and Observation, PCO). A vezérlési és megfigyelési pontok azonosak a szolgálatelérési pontokkal. A legegyszerűbb esetben a vezérlési és megfigyelési pontok közvetlenül a protokollentitás szolgálatelérési pontjai, de két-három réteggel a vizsgálandó protokoll feletti vagy alatti rétegben is elhelyezkedhetnek, sőt egy távoli csomópontban is.

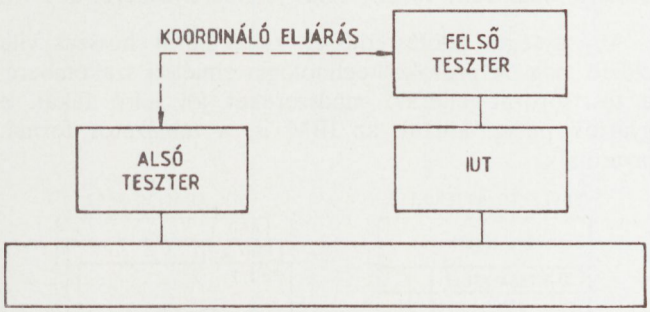
Az alsó és felső teszter

A vizsgáló rendszer elemei
- az alsó teszter,
- a felső teszter és
- a koordináló eljárás.

Az alsó és felső teszter két oldalról határolja a vizsgálandó protokollt (Implementation Under Test, IUT), gyakran a 13. ábrán látható felépítésben.



12. ábra. A vizsgálat elemei: az absztrakt szolgálati primitívek és a protokoll-adatelemek



13. ábra. Az alsó és festő teszter

Az alsó teszter vezérli az absztrakt szolgálati primitívek és a protokoll adatelemek előállítását, megfigyeli és tárolja a vizsgálati kimeneteket. Az alsó teszter funkciói:
- végrehajtja a végrehajtható vizsgálati modulokat,
- létrehozza azokat a protokoll adatelemeket, amelyekre az alsó rétegekben szükség van,
- működteti az alsó rétegek szolgálatait,
- elemzi az alsó rétegből érkező szolgálati primitíveket,
- működteti a koordináló eljárást azért, hogy az IUT alsó és felső határfelületeit szinkronizálja,
- jelentéseket készít a vizsgálandó protokoll viselkedéséről oly módon, hogy ezek más, azonos feltételek mellett történő alkalmasságvizsgálatok eredményével összehasonlíthatók legyenek.

A felső teszter az IUT felső szolgálatelérési pontján és a felette levő rétegek szolgálatelérési pontjain mérhető eseményeket, absztrakt szolgálati primitíveket vezérli és figyeli. A felső tesztert gyakran tesztválaszadónak (test responder) nevezik, mivel gerjeszti azokat a válaszokat, amelyeket az IUT-nek adnia kell. A felső teszter megvalósítása legegyszerűbb esetben az IUT feletti rétegek megvalósítását jelenti.

Az alsó és felső teszter nem működhet egymástól függetlenül, szinkronizálni kell őket. Ez a szinkronizálás közvetlen együttműködést jelent a vizsgálat időtartama alatt, az együttműködés szabályait nevezik koordinációs eljárásnak. A koordinációs eljárás folyamatközi összeköttetéssel vagy tesztmenedzsment protokoll alkalmazásával valósítható meg.

A vizsgálati módszerek osztályozása

Az absztrakt alkalmasságvizsgálati módszereket [21] alapvetően két szempont szerint osztályozzuk:
- hol vannak a vezérlési és megfigyelési pontok,
- hány réteg vesz részt a tesztelésben.

A vezérlési és megfigyelési pontok helye azt jelenti, hogy hol van elhelyezve az alsó és felső teszter. Így négy absztrakt tesztelési eljárást különböztetünk meg:
- helyi tesztelés: vezérlés és megfigyelés is a tesztelendő oldalon,
- elosztott tesztelés: vezérlés a referenciaprotokoll oldalán, megfigyelés a tesztelendő oldalon vagy megfordítva,
- koordinált tesztelés: az elosztott tesztelés továbbfejlesztett változata egy vizsgálati menedzsmentprotokoll beiktatásával,
- távoli tesztelés: vezérlés és megfigyelés is a távoli referenciaprotokoll oldalán.

A másik osztályozási szempont a vizsgálandó rétegekre vonatkozik. Az egyrétegű vizsgálat egyetlen réteg vizsgálatát jelenti függetlenül a többi rétegtől. Elméletileg ez a legvilágosabban megfogalmazható vizsgálat, gyakorlati keresztülvitele azonban nehéz, mert nehéz kiragadni egyetlen réteget a környezetéből. A beágyazott (embedded) vizsgálat egyetlen réteg tesztelését jelenti, de a vezérlési és megfigyelési pontok valamely távoli réteg határfelületén helyezkednek el. A többrétegű vizsgálat valamennyi réteg együttes vizsgálatát jelenti. Ekkor valamennyi protokollt egyetlen protokollnak tekintjük, és a közbenső réteghatárokon sem vezérlést, sem megfigyelést nem végzünk.

5.2. Tesztsorozat generálás

Az itt ismertetendő tesztsorozat-generáló módszerek a protokoll FSM-modelljén alapulnak. Az FSM-implementációk tesztelése az állapotátmenetek ellenőrzését jelenti. Az (s, i, o, s') él teszteléséhez a következő lépések szükségesek [2]:
1. az FSM-implementációt az s állapotban hozni,
2. az i bemenő üzenetet küldeni és megfigyelni az o kijövő üzenetet,
3. megbizonyosodni arról, hogy az FSM-implementáció az s' állapotba (következő állapotba) került.

Az 1. lépés általában csak több üzenetváltással érhető el, a protokollok korlátozott vezérelhetősége miatt csak így állítható be a kívánt állapot. A 3. lépést a protokollállapotok korlátozott megfigyelhetősége teszi nehézzé. A protokollállapotok nem figyelhetők meg közvetlenül, pedig ez fontos a hibák feltárása szempontjából.

A következőkben néhány tesztsorozat-generálási módszert ismertetünk a leggyakrabban használtak közül. Ezek különböznek egymástól abban, hogy:

- milyen tulajdonságokat követelnek meg a protokolltól,
- mely lépéseket teljesítenek a tesztelési lépések közül.

Transition tour (TT) módszer

Ez a módszer a protokollspecifikációban szereplő állapotátmeneteket vizsgálja végig. Előnye az egyszerűsége, azonban a tesztelés 3. lépését (a következő állapotot) nem tartalmazza. Egyes protokollokban létezik a tesztelést könnyítő „állapot" üzenet, melyre adott válasz egyértelműen meghatározza a protokoll állapotát. Ilyen protokollokon a TT módszerrel is teljes tesztelést lehet végezni.

A TT módszerben a cél az, hogy az állapotátmenet gráf minden élét legalább egyszer bejáró minimális hosszúságú utat találjunk. Az e célra szolgáló algoritmusok közül a „Kínai postás" (Chinese Postman) algoritmust ismertetjük. Ez az algoritmus minimális költségű utakat keres egy gráfban. Protokolloknál az élekhez tartozó költségeket egyöntetűen egységnyinek választhatjuk, így az útban szereplő élek számának minimalizálása a cél. Első lépésként új éleket veszünk fel a gráfba oly módon, hogy az új gráf szimmetrikus legyen.

Az így szimmetrikussá tett gráf az Xij-k választása miatt bejárható úgy, hogy minden élen pontosan egyszer megyünk végig, ez az úgynevezett Euler-út. Így a legrövidebb olyan utat kapjuk, mely az eredeti gráf minden élét bejárja.

Megkülönböztető sorozatok módszere

A megkülönböztető sorozat olyan bemeneti üzenetsorozat, mely különböző kimeneti sorozatot ad az FSM-gráf minden állapotára. Gönenc dolgozta ki ezt a módszert, melyben a tesztelés 3. lépése a megkülönböztető sorozattal végezhető el. A módszer előnye, hogy az állapotok megfigyelhetővé tételével nagyon mély tesztelésre ad lehetőséget. Hátránya viszont, hogy csak kevés gyakorlati protokollnak van megkülönböztető sorozata, és ezek általában nagyon hosszúak.

A Gönenc-módszer [12] először ellenőrzi, hogy az FSM-implementáció rendelkezik-e a specifikációban szereplő valamennyi állapottal. Ehhez a szükséges üzenetsorozatokkal eljuttatja az FSM-implementációt az egyes állapotaiba, és mindjárt ellenőrzi a megkülönböztető sorozat segítségével, hogy a kívánt állapotba került-e. Ezután végigvizsgál valamennyi állapotátmenetet. Ennél is a megkülönböztető sorozattal ellenőrzi, hogy az állapotátmenet során jó állapotba került-e az implementáció.

Karakterisztikus sorozatok módszere

Ezt a módszert olyan protokollok esetén használják, melyeknek nincs megkülönböztető sorozata. A karakterisztikus sorozat olyan bemeneti sorozat, amely az s állapotot a többi állapotok egy részhalmazától különbözteti meg. Az ilyen sorozatok teljes halmaza az FSM karakterisztikus halmazát adja, a bemenő sorozatokkal már valamennyi állapotát meg tudjuk különböztetni. Ez a módszer a gyakorlati protokollok esetén jobban használható, mint a megkülönböztető sorozatok módszere, ennek segítségével is elvégezhető a tesztelés 3. lépése.

Egyedi bemeneti/kimeneti sorozatok módszere

Ebben a módszerben egy minimális költségű (hosszúságú) sorozatot határoznak meg minden állapothoz úgy, hogy az erre adott válaszsorozat jellemző az adott állapotra. A megkülönböztető és a karakterisztikus sorozatok arra adtak választ, hogy milyen állapotban van a protokoll; az s állapothoz tartozó UIO-sorozat (Unique Input/Output Sequence) csak arra a kérdésre tud válaszolni, hogy a protokoll az s_i állapotban van-e. Ez a sorozat detektálja a hibát, de nem mondja meg, hogy a várt s_i állapot helyett mely másik állapotba került a protokoll. A gyakorlatban használt protokollokra mégis előnyösebb ez a módszer, mert:

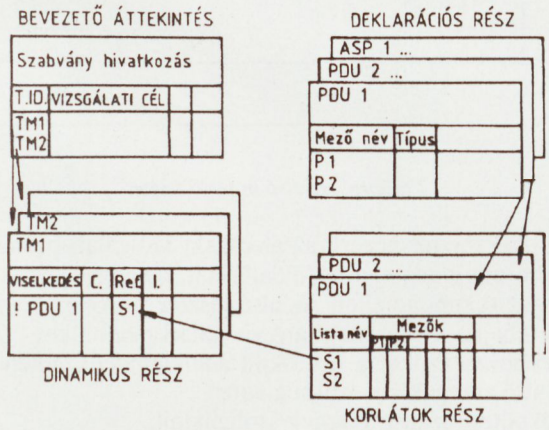
- az UIO sorozat rövidebb, mint akár a megkülönböztető sorozat vagy a karakterisztikus sorozat, hiszen részhalmaza ezeknek,
- majdnem minden protokollnak van UIO sorozata.

A tesztelés 3. lépésének elvégzése céljából egy állapotátmenet először bejárásakor úgy ellenőrizzük, hogy alkalmazzuk rá az állapothoz tartozó UIO sorozatot. Ezért a gráfot ki kell egészíteni tesztélekkel, melyek az UIO sorozat végrehajtását reprezentálják.

A Transition Tour módszerhez hasonlóan a gráfot szimmetrikussá tesszük úgy, hogy eredeti éleiből (tehát nem a tesztélekből) felveszünk még annyit, hogy a kapott gráf az Euler-úton úgy legyen bejárható, hogy közben minden élet pontosan egyszer érintünk. Teszteléskor a tesztélen való áthaladás jelenti az eredeti gráf egy állapotátmenetének és a tesztelés 3. lépésének ellenőrzését, míg az eredeti éleken való áthaladás csak a tesztélek összekötésére szolgál. A tesztsorozatot a kapott gráf Euler-útja adja.

5.3. Az absztrakt tesztkészlet jelölésrendszere: a TTCN

Az egységes jelölésrendszer kialakulását hosszas vita előzte meg. A protokolltechnológia elméleti szakemberei a tesztsorozat-generáló módszereket jól leíró fákat, a gyártók pedig, köztük az IBM is, a táblázatos formát javasolták.



14. ábra. A TTCN táblázatok egymáshoz való viszonya

A tesztfák alapján jól követhetők a tesztelés dinamikus lépései, a táblázatos formával egyértelműbben lehet meghatározni a vizsgálati események (PDU-k, ASP-ok) formátumát. Végül a két irányzatot ötvözték és kialakult a TTCN.

A jelölési rendszer angol [16], [9] elnevezése: Tree and Tabular Combined Notation (TTCN) arra utal, hogy ez a jelölési mód a vizsgálathoz szükséges valamennyi információt viselkedési fák és táblázatok segítségével ábrázolja. A LOTOS-hoz vagy az Estelle-hez képest nem igazán nevezhető formálisnak. Általános célú, létrehozásában a két fő szempont, hogy minden protokollra gazdaságosan alkalmazható legyen és össze lehessen hasonlítani a vizsgálati eredményeket.

Szemben a protokollspecifikációs nyelvek háromféle szabványával a TTCN-t mind az ISO, mind a CCITT elfogadta szabványként. A TTCN-nek grafikus (TTCN-GR) és gépen futtatható (TTCN-MP) változata van.
Egy TTCN leírás négy fő részből áll (14. ábra):

Bevezető áttekintés
Tartalmazza a vizsgálandó protokoll(ok) pontos referenciáját, az alkalmazott tesztelési módszert, a tesztkészlet részletes listáját tesztcsoportonként és minden egyéb a vizsgálat elvégzéséhez szükséges általános információt.

Deklarációs rész
Meghatározza az absztrakt tesztkészlet elemeit és azok paramétereit, a használt adattípusokat a rájuk vonatkozó műveletekkel együtt. Ezekhez tartoznak:
- globális és lokális állandók,
- globális és lokális változók,
- vezérlési és megfigyelési pontok (PCO),
- absztrakt szolgálati primitívek (ASP),
- protokoll adatelemek (PDU),
- időzítők,
- rövidítések.

Dinamikus rész
Táblázatos formában tartalmazza a viselkedési fákat, melyek megadják az egyes PCO-kban megjelenő ASP-ok, illetve PDU-k időbeli sorrendjét és paramétereik pontos értékét (a Korlátok részre való hivatkozással), az egyes ágakhoz tartozó ítéletet, mely a vizsgálat eredményességét határozza meg. Ezek a táblázatok lehetőséget adnak bizonyos mértékű szinkronizációra és alternatív események definiálására is.

Korlátok rész
Megadja a Dinamikus részben használt hivatkozási nevekhez tartozó pontos paraméter-beállításokat a Deklarációs részben meghatározott szolgálati primitívekre és protokoll adatelemekre.

A tesztkészlet legfontosabb elemeit, a tesztmodulokat a Dinamikus rész táblázatai írják le. Minden tesztmodult egy-egy viselkedési fa (tesztfa) ábrázol a hozzá tartozó, a vizsgálatokhoz szükséges kiegészítő információval együtt. Ilyen kiegészítő információk:
- a tesztmodul azonosítója,
- a modul vizsgálati célja,
- az egyes vizsgálati események referenciája, azaz annak a TTCN-leírás Korlátok részében szereplő listanévnek a megadása, amely meghatározza az esemény pontos paraméterértékeit,
- az ítélet, amely a tesztfa adott ágához tartozik és értéke lehet:
  - Pass (sikeres), ha az adott ágban leírt viselkedés a protokollszabályok szerint elvárt vagy megengedett viselkedés;
  - Fail (sikertelen), ha az adott ágon leírt viselkedés az IUT hibájából nem felel meg az elvárásoknak;
  - Inconclusive (következetlen), ha se nem sikeres, se nem sikertelen, azaz az adott események alapján nem tudjuk eldönteni, hogy jó-e vagy rossz az IUT;
- a szinkronizációt biztosító címkék.

Itt érdemes megjegyezni, hogy az ítélet mezőben megadott érték a vizsgálat alatt álló megvalósítást minősíti, tehát nem szabad összetéveszteni egy adott vizsgálat sikerességével, mely a tesztelés menetére vonatkozik. Például, egy adott megvalósításról csak akkor mondhatjuk ki, hogy sikertelen, azaz hibás, ha sikeresen hajtottunk végre egy sikertelen ítélettel záródó sorozatot. Amennyiben nem tudunk végrehajtani egy adott sorozatot, az nem jelenti feltétlenül a megvalósítás hibáját, mivel a tesztkészletek általában nem teljes körű vizsgálatokat tartalmaznak.

A viselkedésleírás lényeges vonása, hogy csak időben egymást követő, illetve alternatív eseményeket tud meghatározni. A TTCN jelenlegi formájában a legnagyobb hátrány, hogy nem ad lehetőséget párhuzamosan zajló események leírására. így nem lehet leírni az alsó és a felső teszter egyidejű működését sem, mivel ez párhuzamos eseményeket is feltételez.

IRODALOM

[1] Bochmann G.v., Vaucher J., "Adding Performance Aspects to Specification Languages" *in: Prot. Specification, Testing and Validation VI.*, North-Holland Publ. Co., 1987.

[2] Bosik S.B., Uyar M.Ü., "Formal Methods in Protocol Conformence Testing: from Theory to Implementation", *Proc. of the IX. Symp. on Protocol Specification, Testing and Verification*, Jun. 6–9. 1989.

[3] Brand D., Zafiropulo P., "Communicating Finite–State Machines", *Journal of the ACM, Vol. 30*, No. 2, April 1983.

[4] Brinskma E., Scollo G., Steenbergen C., "LOTOS Specifications, their Implementations and their Tests", in: *Prot. Specification, Testing and Validation VI.*, North-Holland Publ. Co., 1987. pp.349–360.

[5] Castanet R., Sijelmassi R., "Local Tests for Distributed Testing", in [7], pp.663–676.

[6] Choi T.Y., Miller R.E., "Protocol Analysis and Synthesis by Structured Partitions", *Computer Networks and ISDN Systems 11.* 1986. pp.367–381.

[7] "Computer Network Usage: Recent Experiences", Csaba L., Tarnay K., Szentiványi T. (eds.), North-Holland Publ. Co., Amsterdam, 1986.

[8] Csopaki Gy., "Távközlési szoftver technológia", *Magyar Távközlés* 1992. Maj p. 3-4.

[9] Doubet M.L., "TTCN: Status and Assesment", *2nd International Workshop on Protocol Test Systems*, Berlin (West), Germany, October 3–6. 1989.

[10] Ehrig H., Mahr B., "Fundamentals on Algebraic Specification" Springer Verlag EATCS 6, Berlin 1985.

[11] Fdida S., Wilk A., Perros H., Semaphore Queues: Modelling Window Flow Control Mechanisms, *Report Universite P. et. M. Curie*, No. 181, Paris

[12] Gönec G., "A Method for the Design of Fault Detection Experiments", *IEEE Trans. on Computers*, June 1970.

[13] Gouda M.G., Yu Y.T., "Protocol Validation by Maximal Progress State Exploration", *IEEE Trans. on Communications* Vol. COMM-32, No 1, Jan. 1984.

[14] Harangozó J., "An Approach to Describing a Data Link Level Protocol with a Formal Language", *Fifth Data Comm. Symp.*, 1977 Snowbrid, Utah
[15] Harangozó J., "Protocol Definition with Formal Grammars". *Computer Network Protocols*, A. Danthine (ed.), Liege, 1978.
[16] ISO DIS 9646 "OSI Conformance Testing Methodology and Framework", 1989.
[17] Lee T.T., Lai M.Y., "A Relational Algebraic Approach to Protocol Verification", *IEEE Trans. on Software Engineering*, Vol. 14, No. 2, Feb. 1988.
[18] Losonczi I., "Egy módszer protokollok verifikálására", Diplomaterv, 1989, Budapest.
[19] Milner R., "A Calculus of Communicating Systems", *Lecture Notes in Computer Science*, Vol. 92, Springer Verlag, New York, 1980.
[20] Pehrson V., "Verification of Protocols", *Proc. of Eight International Symposium on Protocol Specification, Testing and Verification*, Atlantic City, N.J. USA, June 7–10, 1988.
[21] Rayner D., "Standardizing Conformance Testing for OSI", in [7], pp.47–68.
[22] Sajkowski M: Protocol Verification Techniques" in: *Prot. Specification, Testing and Validation*, V. North-Holland Publ. Co., 1985.
[23] Sidhu D.P., Leung T.K., "Formal Methods for Protocol Testing: A Detailed Study", *IEEE on SE*, Vol. 15, No 4, 1989.
[24] Tarnay K., "Kommmunikációs protokollok modellezése és konformancia vizsgálata", Doktori értekezés, Budapest, 1990.
[25] Tarnay K., "Protocol Specification and Testing", Akadémiai Kiadó, Budapest, 1991.
[26] West C.H., "General Technique for Communications Protocol Validation", *IBM J.Res. Develop.* Vol. 22, No 4, July 1978.
[27] Zafiropulo P., "Protocol Validation by Duologue-Matrix Analysis", *IEEE Trans. on Communications*, Vol. COM–26, No 8, Aug. 1978.
[28] Vuong, S.T., Hui D.D., Cowan, D.D., "VALIRA – a Tool for Protocol Validation Via Reachability Analysis" in: *Prot. Specification, Testing and Verification VI.*, North-Holland Publ. Co. 1987.

PROTOCOL SPECIFICATION AND TESTING

K. TARNAY

KFKI INSTITUTE FOR MEASUREMENT AND COMPUTING TECHNICS BUDAPEST, PF. 49. H-1525

The communication protocol as a main component of telecommunication software is discussed in this paper. The models based on finite-state machines are most suitable for protocol specification and therefore these are emphasized among the other mathematical models of protocols.

Tarnay Katalin az MTA Mérés- és Számítástechnika Kutató Intézetének tudományos tanácsadója kb. 15 éve foglalkozik kommunikációs protokollok modellezésével és vizsgálatával. Elméleti eredményeiből készült doktori értekezését 1992-ben védte meg, gyakorlati eredményeit a SWIFT nemzetközi bankhálózat alkalmazza. Rendszeresen oktat a Budapesti Műszaki Egyetemen. 1986-ban a Montreal-i Egyetem vendégprofesszora volt. Több mint 70 publikációja jelent meg. 1991-ben az Akadémiai Kiadó és a Plenum Press közös kiadásában jelent meg „Protocol Specification and Testing" c. könyve. 1992. szeptember óta a BME Távközlési és Telematikai Tanszékének címzetes egyetemi tanára.

AZ AXE 10 SZOFTVERTERVEZÉSI FOLYAMATBAN ALKALMAZOTT TECHNOLÓGIA AZ ERICSSONNÁL

SEAD KOTLO

E ICSSON TECHNIKA KFT
110[illegible] UDAPEST, VENYIGE U. 1.

Az AXE 10 digitális kapcsolórendszernél — az Ericssonnak a digitális kapcsolástechnika terén legjelentősebb termékében — alkalmazott szoftvertervezési folyamat több fázisra osztható. Ezek mindegyike több — logikailag egymással kapcsolatban álló — tevékenységből áll. A termelékenység növelésére, és a nemzetközi szabványok által megkívánt minőség biztosítására az Ericsson cég speciális módszereket és hatékony segédeszközöket fejlesztett ki, hogy segítse az AXE szoftvertervezési folyamat különböző fázisait. A cikkben az Ericsson Technika Kft.-nél (Ericsson Technika Hungary – ETH) az AXE szoftvertermékek tervezésére kiépített helyi szoftverfejlesztési környezetet ismertetjük az ott installált segédeszközök fontosabb jellemzőivel együtt, külön kiemelve az alkalmazott szoftvertechnológiát.

1. BEVEZETÉS

Az elmúlt két évtizedben a telekommunikáció területén — különös tekint[illegible]tel a telekommunikációs hálózatokra és azok komponenseire — a fejlesztés intenzívvé vált.

Következésképpen — csak a legnagyobb hálózati komponensre, a kapcsolórendszerre koncentrálva — könnyen felismerhetjük ezt a tényt, és követhetjük a fejlődési lépcsőket, hiszen nyilvánvaló korreláció van e két fejlesztési folyamat között.

A hálózathoz csatlakozó felhasználók folyamatosan egyre fejlettebb és kifinomultabb szolgáltatások iránti követelményekkel lépnek fel. A hálózat szintjén ez a meglévő szolgáltatások szinte állandó bővítését eredményezte, mely főként új funkciók — például a kapcsolási pontok — bevezetését jelentette a kapcsolórendszerekbe.

Az új funkciók megvalósításának módja nagymértékben függ a bevezetés időpontjában elérhető technológia szintjétől.

Más szavakkal, a kapcsolórendszerekkel kapcsolatos kívánságok szintjén a felhasználók definiálják a funkcionális tulajdonságokat, míg a technológia szintje határozza meg a kivitelezés módját.

Amikor az új funkciók beépülnek a kapcsolórendszerekbe — például a kapcsolási pontok —, a felkínált szolgáltatásokat tekintve a hálózat globális potenciálja megnő. A szolgáltatás „térfoglalását" követően, miután a felhasználók megszokják az újonnan felkínált szolgáltatást, új követelmények fogalmazódnak meg, amelyek új szolgáltatásokat igényelnek. Ezen elvárások teljesítésére a hálózat kezelői arra kényszerülnek, hogy bővítsék a meglévő funkciókat, és a fentebb vázolt ciklus kezdődik újra.

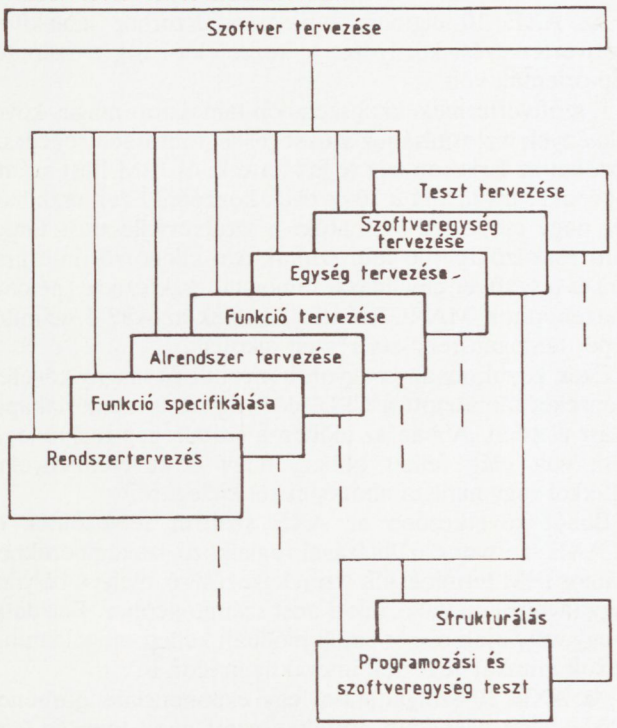
Valójában ez egy végtelen folyamat, mely a telekommunikációs hálózatnak és minden komponensének fokozatos fejlődését eredményezi. A globális hálózati potenciál szintén folyamatosan bővül.

Ilyen állandóan változó környezetben csak a legflexibilisebb és a „legjövőbiztosabb" hálózati komponensek használhatók, amelyek hosszú ideig tökéletesen kielégítik a funkcionalitásra, a minőségre, a fenntarthatóságra és a megbízhatóságra vonatkozó követelményeket. Ezért, ami a struktúráját és az architektúráját illeti, az AXE 10 digitális kapcsolórendszer a kezdetektől fogva teljesen a modularitás koncepcióján alapul, a funkcionalitás, a technológia, a szoftver és a hardver szempontjából egyaránt.

E ténynek nagy hatása van az AXE 10 fejlesztési modellre, mely a különféle fejlesztési folyamatokban — köztük a szoftvertervezési folyamatban — alkalmazott módszertant jellemzi.

Ez a modell a módszertan további fejlesztésének kiindulópontjaként szolgálhat.

A rendszer moduláris struktúráját követve, a szoftvertervezési folyamat maga is számos folyamatra osztható, amelyek mindegyike több különféle tevékenységet foglal magában, amint az az 1. ábrán látható. Az AXE 10 fejlesztési modellnek megfelelően minden ilyen folyamat pontosan definiált és a megfelelő „Folyamatleírás" dokumentumban van rögzítve, amelyek teljes halmaza reprezentálja az AXE 10 szoftvertervezési módszertant.



1. ábra

A folyamatok közötti pontosan definiált interfészeknek — például a megkívánt bemenetek dokumentációja egy folyamat kezdetén és a várható kimenetek dokumentációja

annak eredményeképpen —, az elérhető adatbázisoknak és könyvtáraknak, ahonnan kiveszik, illetve ahol tárolják az adatokat és a dokumentumokat; a segédeszközöknek, amelyeket bizonyos tevékenységek elvégzéséhez használnak; és más fontos információknak a leírását tartalmazza a „Folyamatleírás" dokumentum.

A szoftvertervezési folyamatra vonatkozó „Folyamatleírás" dokumentumok összessége definiálja a fejlesztő környezettel kapcsolatos követelményeket.

A telekommunikáció területén történő szoftverfejlesztés — különösen az SPC kapcsolórendszerek esetében —, mely egyike a legbonyolultabb valós idejű automatáknak, nagyon magas megbízhatóságot, vagyis minőségi terméket követel.

Az ISO 9001-beli kívánalmak egyike, hogy a kapcsolórendszert fejlesztő cégeknek jól dokumentált, jól ismert és konzisztens rendszerekkel kell rendelkezniük mind a hardvert, mind a szoftvert illetőleg. A szoftvert illetően e kívánalmaknak kritikus fontossága van, és csak professzionálisan szervezett szoftvertervezési folyamattal érhető el.

Csak a megfelelő módszertan és a megfelelő környezet tudja biztosítani a szoftvertermékek magas minőségét.

Az AXE 10 fejlesztési modell és az AXE 10 szoftvertervezési módszertant definiáló „Folyamatleírás" tökéletesen teljesíti a fenti minőségi követelményeket a nemzetközi szabványoknak megfelelően, növelve egyben a termelékenységet is.

Az Ericsson Technika Kft.-nél kiépített, az AXE 10 szoftvertervezési folyamatot támogató szoftverfejlesztési környezet leírását adják a következő fejezetek.

2. A FEJLESZTŐ KÖRNYEZET

Az AXE 10 digitális kapcsolórendszerhez kialakított szoftvertervezési környezet a kezdetektől fogva számítógép-orientált volt.

A szoftvertermékekkel szemben támasztott magas követelmények biztosításához szükséges legfontosabb segédeszközöket az Ericsson cég fejlesztette ki és IBM host számítógépekre installálta a 70-es évek közepén. Ezen eszközök két nagy csoportra oszthatók: a szoftverfejlesztést támogató eszközökre (például szintaktikai ellenőrző, interpreter) és a szoftver előállítását támogató eszközökre (például kódgenerátor, MARGEN stb.), melyek az APS 2 számítógépes támogatórendszer részeit alkotják.

Ezek közül néhány nagyon bonyolult és magas követelményeket támasztott a CPU időt vagy/és a memóriakapacitást illetően. Abban az időben a számítógép-technológia nem volt elég fejlett ahhoz, hogy e követelményeket PC-kkel vagy miniszámítógépekkel kielégítse.

Ebből következően az AXE szoftver tervezőinek és az AXE szoftver előállításával foglalkozó szakembereknek számos IBM terminál állt a rendelkezésére, melyek helyileg vagy távolról csatlakoztak a host számítógéphez. Feladataik egy vagy csak részét batch módban kellett megoldaniuk, de sok munkát végeztek interaktív módon is.

Az AXE 10 szolgáltatásai egy exponenciális görbének megfelelően állandóan nőttek, s ezért mind több és több humán erőforrást foglalkoztattak a szoftverfejlesztés területén. Mivel minden felhasználó egy vagy csak nagyon korlátozott számú host számítógéphez volt csatlakoztatva, ez nagyon gyakran túl hosszú válaszidőt okozott.

Másrészről a PC-k és a miniszámítógépek technológiájának területén végbement gyors fejlődésnek köszönhetően nagyon vonzó és egyre hatásosabb termékek jelentek meg a piacon.

E tényeket szem előtt tartva, az Ericsson cégen belül azt a döntést hozták, hogy a nagyszámítógépes támogató rendszer (az APS) fejlesztésének folytatása mellett — mely később az új, továbbfejlesztett APS 3-at eredményezte —, azzal párhuzamosan elkezdték egy PC-orientált, szoftvertervezést támogatórendszer fejlesztését is.

Évekkel ezelőtt ez a fejlesztés eredményezte az első verzióját egy olyan, a szoftvertervezést támogatórendszer megjelenésének, melyet teljesen az IBM host számítógépes környezeten kívül lehetett megvalósítani egy néhány szervert tartalmazó, különálló munkaállomásokból álló hálózatban. Ez az első verzió — melyet APStools-nak hívtak — a meglevő tervezési segédeszközöknek csak a leggyakrabban használt néhány elemét tartalmazta.

Természetesen szükségessé vált, hogy kapcsolatot teremtsenek a hálózati szerver és az IBM host számítógép között, hogy elérjék az IBM számítógépen tárolt nagy adatbázisokat és könyvtárakat — ezekre a különféle szoftvertervezési tevékenységek miatt volt szükség —, valamint hogy lehetővé tegyék a csak az IBM környezetben létező eszközök használatát.

Mivel az első gyakorlati tapasztalatok nagyon pozitívak voltak, a fejlesztés folytatódott és új APStools verziók jelentek meg, melyek egyre több lehetőséget és funkciót tartalmaztak, az előző verziónál fejlettebb formában.

Valójában a csúcstechnológiát képviselő termékeket alkalmazva, s azok nagy technikai teljesítőképességét kihasználva az AXE szoftvertervezési környezetnek egy olyan változata készült el, mely egyre függetlenebbé vált az IBM host környezettől. E folyamat nemsokára teljesen befejeződik, néhány központi adatbázis és közös könyvtár kivételével, amelyek a host számítógépen maradnak, hogy megkönnyítsék azok folyamatos fenntartását és naprakész állapotban tartását. Ilyen körülmények között ez év kezdetére egy, a fentebb vázoltnak megfelelő tervező környezet valósult meg és lépett működésbe az ETH-nál, az Ericsson magyarországi vállalatánál.

Hardver szempontból ezt a kezdeti kapacitásra méretezték ugyan, de teljesen nyitott a későbbi bővítések számára, mely a szoftvertervezés és az ETH részvételével zajló fejlesztési munkák területén lekötött emberi erőforrásoknak a tervek szerinti dinamikus növekedését követi majd.

Ami az APStools-t, vagyis a szoftvert illeti, az a legutolsó verziót tartalmazza, mely — sok egyéb más előny mellett — az IBM környezet majdnem teljesen funkcionalitását nyújtja.

Az egyszerűsített hardverstruktúra a 2. ábrán látható.

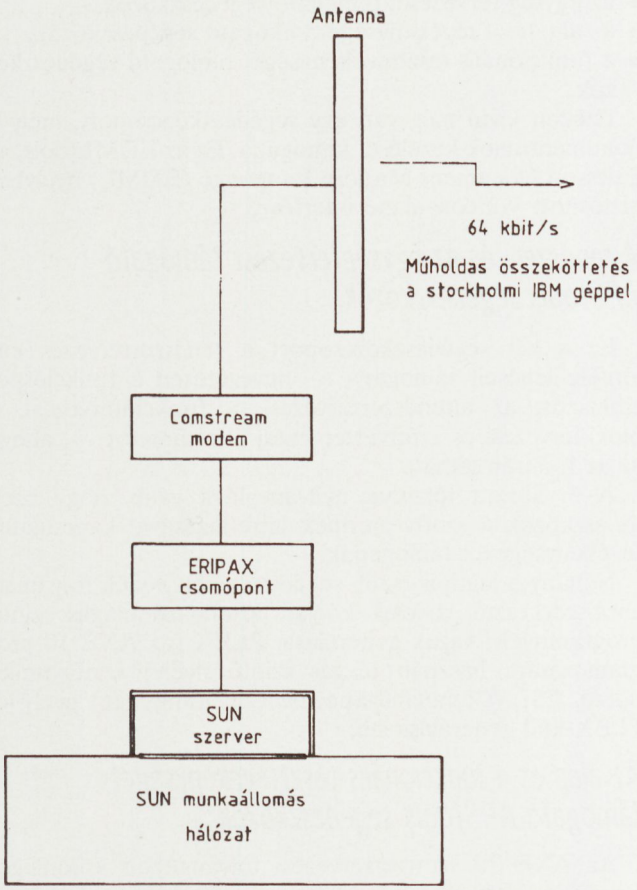
A munkaállomásokat és a szervert összekötő lokális hálózat egy árnyékolatlan, kéteres, sodrott kábeles ETHERNET hálózat. Az Access Hub-on alapuló csillagtopológia biztosítja a nagy flexibilitást, a lehetséges további bővítést, például a más lokális hálózattal való összeköthetőséget.

Mind a szerver, mind a munkaállomások a SUN Microsystems által 1991. végén kínált termékeknek megfelelők.

A SPARCserver 490 jellemzői:

- VME busz
- 33 MHz-es SPARC CPU 19.4 SPECmarks (22 MIPS; 3,8 MFLOPS)
- 33 MByte ECC memória
- 3 × 900 MByte IPI diszk

- MCP (Többcélú Kommunikációs Processzor) kártya
- CD ROM egység
- 1/4" streamer 150 MByte
- 8 mm Exabyte streamer 2,3 GByte



2. ábra

A Winchester nélküli munkaállomások SUN/ELC típusúak és jellemzőik a következők:

- 33 MHz-es SPARC CPU 17.1 SPECmarks (21 MIPS, 3 MFLOPS)
- 12 MByte RAM
- 19" monokróm monitor

Mind a szerver, mind a munkaállomások konfigurációja azokat a követelményeket tükrözi vissza, amelyeket az AXE szoftvertervező környezet állított fel.

Az installált operációs rendszer SunOS 4.1.1. típusú, a SUN által továbbfejlesztett 4.3 BSD UNIX rendszer. A grafikus felhasználói interfész (Graphical User Interface – GUI) Open Look alapú Open Windows. A UNIX fejlett hálózati komponensei, például az NFS, a TCP/IP és a SUN által szabadalmaztatott Hálózati Információs Szolgáltatás (Network Information Service – NIS) nagyon hatékonnyá teszik a Winchester nélküli munkaállomások használatát.

A szerveren futó SunLink SNA 3270 szoftver az IBM 3274-es Model 1C cluster vezérlőt emulálja, és 32 egyidejű terminálemulátor működéséhez nyújt támogatást. A számítógépes munkaállomás szoftverkomponense az IBM 3278 képernyőterminál és a 3287 nyomtató emulációját valósítja meg. A nyomtató kimenete UNIX rendszer file-ba, illetve eljárásba is átirányítható. A hálózati nyomtatást PostScript lézernyomtató támogatja.

Mind a szerver, mind az adatkapcsolati berendezés nagyteljesítményű, szünetmentes áramforrásra (Uninterruptable Power Supply Unit – UPS) van csatlakoztatva. Egy megfelelő program vigyázza a szerver biztonságos leállítását hálózati hiba esetén.

Az ETH-ban telepített SUN hálózat 1 szervert, 15 munkaállomást, 1 QMS lézernyomtatót, 1 UPS egységet, 1 Access Hub-ot és árnyékolatlan, kéteres, sodrott kábelt tartalmaz. A rendszer teljesen nyitott a jövőbeli bővítésekre.

Az adatkapcsolati berendezés lehetővé teszi az EUTELSAT II-n keresztüli műholdas összeköttetést az Ericsson stockholmi központi számítógépeivel. Az ETH-beli SUN felhasználóknak ezen keresztül nemcsak az Ericsson/Stockholm adatbázisaihoz és könyvtáraihoz van távoli hozzáférési lehetőségük, hanem azokhoz is, amelyek az Ericsson más vállalatainál és leányvállalatainál találhatók a világ egyéb részein, aszerint hogy az ETH fejlesztői milyen fejlesztési munkákban vesznek részt, amelyeket a Svédországon kívüli Ericsson válallatok vezetnek vagy koordinálnak, hiszen azok is össze vannak kötve egymással hasonló módon.

A 2. ábrán látható adatkapcsolati berendezés a következő részekből áll:

- 2,4 m-es parabola antenna, LNC konverterrel és megfelelő teljesítményerősítővel,
- Comstream modem,
- ERIPAX csomagkapcsolt csomópont több kis- és nagysebességű porttal, mely többféle protokollt és interfészt támogat, például X.25/75-öt, SDLC-t, V.24-et a szinkron és aszinkron adattovábbításhoz.

Az ETH-beli SUN hálózat és az Ericsson/Stocholm IBM számítógépe közötti műholdas összeköttetés pont–pont típusú. Valójában ez egyfajta bérelt összeköttetés, mely földi összeköttetés helyett műholdat használ.

Az átviteli sebesség 64 kbit/s, amely a kezdetben várt adatforgalomhoz elégséges. Ha és amikor szükségessé válik, a későbbiekben ez az átviteli sebesség növelhető.

A SUN hálózat mellett az ETH — különálló PC-s lokális hálózat által megvalósított — Információs Rendszere is csatlakozik az ERIPAX csomóponthoz, biztosítva az adatkapcsolathoz való hozzáférést, s ezáltal a globális Ericsson Információs Rendszer integráns részévé válik.

3. AZ AXE 10 SZOFTVERTERVEZŐ ESZKÖZÖK

Az AXE 10 digitális kapcsolórendszer számára történő szoftverfejlesztés az általános célú számítógépeket szolgáló hagyományos szoftverfejlesztéstől lényegesen eltér a következők miatt:

- az AXE 10 digitális kapcsolórendszert speciális, dedikált számítógép vezérli, melynek processzorait szigorú valós idejű igények kielégítésére tervezték;
- az AXE 10 digitális kapcsolórendszer szoftverének ki kell elégítenie a globális rendszermodularitás-koncepción alapuló követelményeket, azaz modulárisnak kell lennie;
- a szoftvernek jól strukturáltnak és nemcsak a tervezők, hanem a fenntartó személyzet számára jól érthetőnek kell lennie;
- a szoftvernek a legprofesszionálisabb minőségűnek kell lennie.

Következésképpen mindezeket a tényeket és elvárásokat figyelembe vették a szoftvertervezést támogató rendszer kifejlesztésénél, amely számos hatékony és kifinomult tervezési segédeszközt tartalmaz.

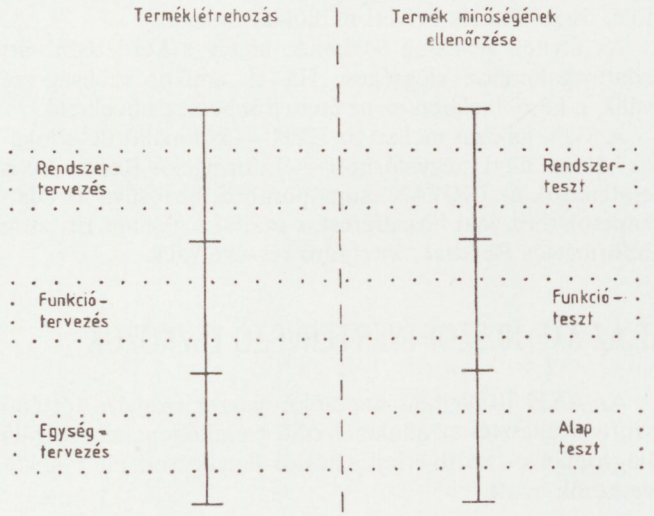
Ahogy azt az előző fejezetben említettük, az AXE 10 szoftvertervezési segédeszköznek az IBM-ről nem-IBM környezetre való áttérése egyre intenzívebbé vált az idők folyamán. Ma ez a folyamat majdnem befejeződött az APStools programcsomag kifejlesztésével, amelyet különböző, SUN termékeken alapuló, különálló hálózatban használnak az AXE 10 szoftvertermékeinek fejlesztésére.

Az APStools támogató környezet számtalan Window-alapú objektum-orientált segédeszközt nyújt. Az APStools felhasználói interfésze objektum-orientált megközelítésű.

Egy objektum lehet egy — hagyományos értelemben vett — file, egy forrásprogram-dokumentum, egy leírás vagy egy összetett objektum, mint például egy dosszié vagy egy szoftverkomponens.

A felhasználói interfész működése a „select-and-act" elven alapul, azaz egy művelet a kiválasztott objektumon azonnal, rögtön a kiválasztás után végrehajtódik. A felhasználói interfész felismeri a kiválasztott elem típusát és képes annak eldöntésére, hogy melyik specifikus műveletet kell elvégezni a kiválasztott általános művelet keretében. Egy strukturált objektum, mint például egy dosszié megnyitása után megjelenik annak tartalma, míg például egy dokumentum megnyitása után az annak áttekintéséhez és megvalósításához szükséges eszközök aktivizálódnak.

Az objektum-modell direkt módon képződik le a UNIX hierarchikus file-rendszerére. A strukturált objektumok — mint például egy dosszié — egyszerűen könyvtárak a file-rendszerben. A dokumentumok és a forrásdokumentumok közönséges file-ok. Következésképpen az objektumokra vonatkozó elnevezési szabályok szigorúan követik a UNIX file-rendszer elnevezési szabályait.



3. ábra

Ugyanúgy, mint a file-rendszerben, egy objektum meghatározható a file-rendszer gyökeréből kiinduló teljes elérési úttal vagy az aktuális környezethez képesti relatív elérési úttal.

A 3. ábra — mely csak a fontosabb fázisait ábrázolja a szoftvertermékek létrehozásának és verifikálásának — és az 1. ábra — melyen az AXE 10 szoftvertervezési folyamat struktúrája látható — összevetésével könnyen felfedezhetjük a kapcsolatot — a különböző részfolyamatokon belüli tevékenységek támogatását illetően — ezek és a kifejlesztett APStools között.

Az APStools-on belül négy fő csoport található:

- a rendszertervezést támogató segédeszközök,
- az egység tervezését támogató segédeszközök,
- az alapteszt tevékenységet támogató segédeszközök,
- a funkcionális teszttevékenységet támogató segédeszközök.

Ezeken kívül még van egy segédeszközcsoport, mely a dokumentumok kezelését támogatja. Ez az EDMLtools, az Ericsson Document Markup Language (EDML) nyelvhez biztosított Window-alapú interfész.

A rendszer- és az egységtervezési támogató APStools segédeszközök

Ez a két segédeszközcsoport a rendszertervezés különféle lépéseit támogatja — nevezetesen a funkcióspecifikációs, az alrendszertervezési, a funkciótervezési, a blokktervezési és szoftvertervezési alfolyamatot —, ahogy az az 1. ábrán látható.

A 3. ábrára tekintve nyilvánvalóvá válik, hogy ezek az eszközök a szoftvertermék létrehozásával kapcsolatos tevékenységeket támogatják.

Néhány a legtipikusabb segédeszközök közül: folyamatábra-szerkesztő, a jelek kódját tartalmazó magas szintű programnyelvi vázak generálása, PLEX (az AXE 10 programozására használt magas szintű nyelv) szintaxisellenőrző, CSL (Command Specification Language) nyelvből PLEX-kód generálása stb.

Az alap és a funkcionális teszttevékenységet támogató APStools segédeszközök

Az AXE 10 szoftvertervezési folyamatban különleges figyelmet fordítottunk a szoftvertermékek minőségére. A lehetséges legmagasabb szint elérésére a szoftvertermékek tesztelésének speciális koncepcióját alakítottuk ki, amely három különböző szintjét — követve az AXE 10 rendszer struktúráját — biztosítja a minőség ellenőrzésének: alapteszt, funkcionális teszt és rendszerteszt. Ez a 3. ábrán mint a termék minőségének ellenőrzése jelenik meg.

A funkcionális teszt az AXE 10-ben a szoftvertermékek minőségi ellenőrzésének egy magasabb szintjét testesíti meg. Ezt az alapteszt tevékenységek sikeres befejezése után hajtjuk végre. Az alapvető különbség a funkcionális teszt és az alapteszt között, hogy az előbbiben a terméket a valóságos környezetében, azaz egy System Test Plantnek (STP) nevezett AXE 10 kapcsolórendszer modellben tesztelük.

A rendszerteszt még inkább valóságoskörnyezet-orientált, ez nemcsak a kiválasztott funkció, hanem a megfelelő rendszerszint teljes funkcionalitásának tesztelésére szolgál, ezért egy APStools-on kívüli, más segédeszköz támogatja.

Néhány a legtipikusabb alap- és funkcionális tesztelést támogató eszközök közül: a folyamatábra összes lehetséges útvonalát bejáró tesztesetek vázának generálása, mind a magas szintű (PLEX), mind az assembly szintű (ASA/C, R) nyelven írt programok hibáinak feltárására szolgáló interaktív tesztelés, CPU szoftver dinamikus tesztelése az AXE központi processzorának (APZ 212) emulálásával végrehajtva, kommunikációs kapcsolat a SUN/UNIX hálózat és az AXE 10 kapcsolórendszer input/output alrendszere között stb.

4. ÖSSZEFOGLALÁS

Az Ericsson Technika Kft.-nél (ETH) kiépített tervezési környezet teljesen összhangban van az Ericsson cég által az AXE 10 digitális kapcsolórendszer szoftvertervezési folyamatára vonatkozó legutolsó ajánlással. Hardverkonfigurációja lehetővé teszi a szükségleteknek és a terveknek megfelelő későbbi bővítést, és megteremti a szükséges hozzáférést a teljes Ericsson cég adatbázisaihoz és könyvtáraihoz. Szoftverét tekintve, az ETH szoftvertervezőinek több hatékony, kifinomult segédeszközt nyújt, lehetővé teszi a felhasználóközeli, kényelmes környezetben való munkát, hogy a szoftvertermékeket a legprofesszionálisabb minőségben állítsák elő, munkájuk minden fázisában szigorúan azokat a rutinokat és módszereket alkalmazva, amelyeket az AXE szoftverfejlesztési módszertan definiál.

Emellett az ismertetett szoftvertechnológia nagymértékben, legalább 30–40%-kal növeli a produktivitást e területen, s nemcsak hogy nem veszti el, hanem fokozza a szoftvertermékek minőségét.

SOFTWARE TECHNOLOGY APPLIED AT THE ERICSSON WITHIN THE AXE 10 SOFTWARE DESIGN PROCESS

S. KOTLO

ERICSSON TECHNIKA LTD., VENYIGE U. 1. BUDAPEST, H-1108

Software design process for the digital switching system AXE 10, the main Ericsson's product in field of digital switching, is divided in a number of subprocesses/phases. Further on, each of them includes a number of activities logically related to each other. In order to increase the productivity as well as to ensure the quality required by international standards, special methods and powerful tools have been developed by Ericsson company supporting different phases within the AXE software design process. The local software design environment built up at Ericsson Technika Ltd. for design of AXE software products together with brief characteristics of the installed tools are discussed pointing out applied software technology.

Sead Kotlo 1966-ban szerzett diplomát a Zágrábi Egyetem Elektrotechnikai Karán, ugyanitt kapott MS fokozatot 1982-ben a távközlés témakörében. 1967-től a zágrábi Nikola Tesla távközlési vállalat kutatórészlegében dolgozott, ahol kapcsolórendszerek fejlesztésével foglalkozott. 1978-tól az AXE rendszer szoftverfejlesztésével, 1981-től a teljes AXE 10 rendszer fejlesztésével foglalkozott a Tesla vállalatnál. 1991-ben a magyar Ericsson Technika Kft. szoftverfejlesztési részlegének vezetője lett. A Kft. az Ericsson általános AXE 10 fejlesztési programjában is részt vesz.

AZ OBJEKTUM-ORIENTÁLT PROGRAMOZÁSI TECHNIKA ALKALMAZÁSA TÁVKÖZLÉSI SZOFTVEREK ELŐÁLLÍTÁSÁRA

ADAMIS GUSZTÁV

TÁVKÖZLÉSI ÉS TELEMATIKAI TANSZÉK
BUDAPESTI MŰSZAKI EGYETEM
H-1111 BUDAPEST, STOCZEK U. 2.

A cikk az objektum-orientált programozásnak a távközlési szoftverekre gyakorolt hatását elemzi. Bemutatja a távközlési szoftverek speciális követelményeit, az objektum-orientált programozás legfontosabb ismérveit és azt, hogy a modellalkotási folyamat hogyan tükröződik az objektum-orientált programozás elveiben. Végezetül összefoglalja az objektum-orientált programozás (és különösen a C^{++} nyelv) előnyeit a távközlési szoftverek fejlesztésében.

1. A TÁVKÖZLÉSI SZOFTVEREK JELLEMZŐI

Mai világunkban robbanásszerű fejlődés megy végbe a telekommunikáció területén. Mind újabb és újabb berendezések, szolgáltatások jelennek meg a piacon, a távközlési eszközök döntő többsége valamilyen módon kapcsolatban áll digitális számítógépekkel. Ezért nem meglepő, hogy szinte exponenciális sebességgel nő az igény a távközlési szoftverek iránt.

A távközlésben használt szoftverek általában speciális követelményeket támasztanak. Ezek közül a fontosabbak:

- Nagy megbízhatóság.
  E szoftverek jelentős hányada folyamatos működésben üzemel, ahol nem megengedhető, hogy egy esetleges hiba miatt akár rövid időre is megszakadjon a működés. Ezért az ilyen szoftvertermékekben nagy gondot kell fordítani a hibák számának minimalizálására (ez a tevékenység az ilyen szoftver kifejlesztési idejének és költségének a felét is jelentheti) és az esetleges hibák hatásának minimalizálására (amely kívánalom speciális programozástechnikai igényeket vet fel).
- Könnyű fenntarthatóság.
  E kívánalom szellemében biztosítani kell a program könnyű áttekinthetőségét, strukturáltságát és érthetőségét nemcsak a program tervezője, hanem a program használója számára is.
- Könnyű bővíthetőség.
  Szükséges, hogy a program bővítése, változtatása minél kevesebb módosítással járjon a megmaradó részekben. Ideális esetben a program bővítése a meglevő kód megváltoztatása nélkül, pusztán új részek hozzáadásával jár.
- Speciális felhasználói igények kielégítése.
  Ez a meglévő „alapprogram" speciális kiegészítéseit, módosításait jelenti. E szempontból kulcsszerepe van a modularitásnak.
- Nagy méretek.
  A távközlési szoftverek általában meglehetősen nagy méretűek, ezért egy szoftver kifejlesztésén több programozó dolgozik. Szükség van arra, hogy az egyes programozók által megírt részek között jól meghatározott, csak engedélyezett módon átjárható interfészeket tudjunk definiálni.
- Nagy sebesség.
  Sok alkalmazásban a gyors futási sebesség, a rövid válaszidők elérése fontos kívánalom.
- Rövid fejlesztési idő.
  Ma általános tendencia, hogy a hardvereszközök fejlesztése messze megelőzi a hozzájuk kifejlesztett szoftverek fejlesztési sebességét. A fejlesztési idő csökkentése érdekében biztosítani kell, hogy a kifejlesztendő szoftver szerkezete minél nagyobb mértékben kövesse a megoldandó probléma fizikai sajátosságait, és a programozási nyelv minél inkább takarja el a számítógép hardverjellemzőit, igényeit (ez a forrásprogram hordozhatóságát is növeli), minél közelebb álljon a természetes emberi gondolkodásmódhoz.
- A fejlesztési költségek minimalizálása.
  Ez a kívánalom magában foglalja a program kifejlesztési idejének csökkentését, a hibakeresés egyszerűsítését, a könnyű módosíthatóságot.
- Hordozhatóság.
  Mivel az előbbiek értelmében egy távközlési szoftver kifejlesztése általában hosszú ideig tart és drága, ezért ésszerű kívánalom, hogy a kifejlesztett szoftver ne csak egy fizikai környezetben legyen felhasználható, futtatható. Ez a követelmény egyrészt a hierarchikus programfelépítést követeli meg, vagyis azt, hogy a programnak csak bizonyos alsóbb szintű eljárásai kötődjenek a hardverhez és az operációs rendszerhez, és az e szint által nyújtott szolgáltatásokat felhasználva, a felsőbb szintek minél kisebb mértékben függjenek a fizikai környezettől. Ha a programot át akarjuk vinni egy másik rendszerre, csak ezeket az alsóbb szintű eljárásokat kell megváltoztatni. Ez a kívánalom is megköveteli a modularitást, a jól definiált interfészek meglétét, de általában a sebesség csökkenését is okozza.

A fentebb felsorolt kívánalmak teljesítését a ma elterjedt programozási nyelvek csak többé–kevésbé teszik lehetővé, többük megvalósítására csak nagy nehézségek árán van lehetőség, általában egy másik kívánalom nagymérvű

megsértésével. Ezért volt nagy szükség olyan programozási nyelvek megjelenésére, amelyek kifejezetten támogatják a fenti kívánalmak egyszerű teljesítését, s amelyeket nagyméretű, megbízható, moduláris, könnyen bővíthető, viszonylag gyors és gyorsan elkészíthető szoftverek készítésére optimalizáltak. A programnyelvek ezen új generációja az objektum-orientált programozási nyelvek családja.

2. A MODELLALKOTÁS LÉPÉSEI

Az objektum-orientált programozási nyelvek mögött egy új, a hagyományos strukturált programozástól teljesen eltérő filozófia húzódik meg.

Míg a strukturált programozás mottója a „program végrehajtása (részlegesen sorrendezett) függvényhívások sorozata, amelyben a függvények adatstruktúrákat manipulálnak" [1], addig az objektum-orientált programozás mottója a „program egy fizikai modellnek tekinthető, mely a világ egy valóságos vagy képzeletbeli részét szimulálja" [1].

Mit is jelent ez a valóságban?

Fizikai modellalkotásunk során igyekszünk megkeresni a valóságnak bizonyos önálló jelentéssel bíró, jól körülhatárolható egységeit, objektumait. Meghatározzuk ezen objektumok állapotát leíró, valamilyen formában mérhető jellemzőit, amelyek értékeinek együttes ismerete jellemzi az adott objektum (statikus) állapotát. Ezen kívül meg kell határoznunk, hogy az adott objektum milyen kapcsolatban áll a külvilággal. Egyrészt, hogy hogyan reagál a kívülről érkező hatásokra — ennek során meg kell határoznunk, hogy milyen esetleges változások következnek be az adott objektum állapotában —, másrészt, hogy az adott objektumok — akár a külvilágból érkező hatások eredményeképp, akár spontán módon — milyen befolyással lesznek más objektumokra. E tevékenységet röviden a dinamikus viselkedés leírásaként, illetve az objektum és a külvilág közötti határfelület, interfész leírásaként jellemezhetjük.

Ha az adott objektum leírását befejeztük, akkor hozzákezdhetünk modellünk finomításához.

Egyrészt már meglévő, általános objektumainkat kiegészíthetjük bizonyos speciális tulajdonságokkal, (specializáció). Például „a trolibusz olyan autóbusz, amely felsővezetékhez kötött". Figyeljük meg, hogy az ilyen definíciók során nem soroljuk fel újra a speciális objektum összes tulajdonságát, hanem csak a különbségeket ahhoz az objektumhoz képest, amelyből leszármaztattuk; és hallgatólagosan tudomásul vesszük, hogy az itt felsorolt speciális tulajdonságokon kívül minden olyan jellemzővel is rendelkezik, amellyel a „szülője".

Hasonló jellegű, csak ellentétes irányú tevékenység az absztrakció. Ennek során egy vagy több „speciális" objektumból hozunk létre egy általánosat, segítve a modellalkotást bizonyos túl részletező tulajdonságok elfedésével. Előző példánknál maradva így hozhatjuk létre a személyszállító jármű kategóriáját például a személyautó, autóbusz, trolibusz fogalmakból.

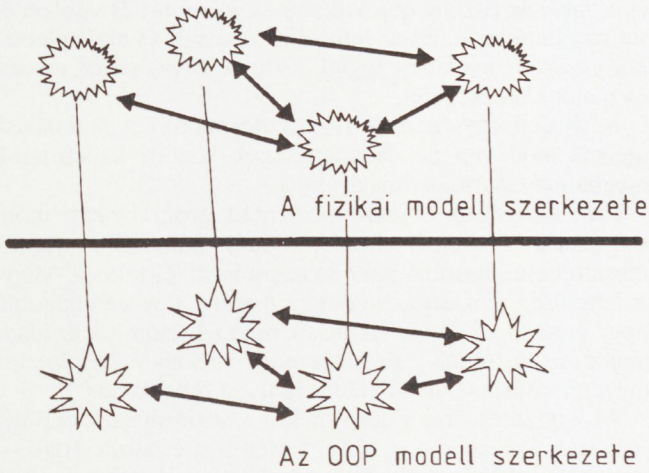
A modellalkotás másik fontos eljárása az összefogás (aggregáció). Ennek során már meglévő különböző objektumok összefogásával alakítunk ki egy új, magasabb minőségi szintet képviselő összetett objektumot. Vagyis példánkban azt mondhatjuk, hogy a motor, a karosszéria stb. alkotja az autót.

Hasonló jellegű, csak éppen ezzel ellentétes irányú tevékenység a dekompozíció. Ennek során egy már meglévő objektumban keresünk és különítünk el funkcionálisan önálló egységeket.

Az ismertetett modellalkotási módszerek kombinált használatával tetszőlegesen bonyolult, összetett, többszintű (hierarchikus szervezésű) modelleket alkothatunk, amelyek a valóságot az általunk megkívánt mértékben jellemzik.

Az objektum-orientált programozás és a modellalkotás kapcsolata

Az objektum-orientált programozási módszer pontosan az előbb vázolt modellalkotási folyamaton alapszik. A módszer szerint elkészített programokban a programozási egységek a lehető legnagyobb mértékben megfelelnek fizikai modellbeli ekvivalensüknek (1. ábra). Így nincs szükség arra, hogy a programírónak kelljen bonyolult és sok hibalehetőséget magában rejtő módon áttranszformálnia a fizikai modellben szereplő jellemzőket a számítógép által megkívánt aktív (függvény) és passzív (adat) egységekbe.



1. ábra. Az objektum-orientált programozási (OOP) modell hűen követi a fizikai modell szerkezetét

Az objektum-orientált programozásban szereplő alapegység az osztály (class), mely megfelel fizikai modellünk önálló jelentéssel bíró, jól körülhatárolt egységeinek. Az osztály — más néven absztrakt adattípus — egyes konkrét egyedei, példányai, „változói" az objektumok.

Az osztályok magukban foglalják az adott fogalom összes, tehát passzív és aktív jellemzőit egyaránt.

Tartalmazzák az objektumok állapotát leíró változókat, melyek elérési (olvashatósági/írhatósági) jogosultságának meghatározására is lehetőség van. Vagyis nagyon egyszerű mód van arra, hogy bizonyos — állapotinformációkat tartalmazó — adatokat elrejtsünk a külvilág elől, és azok megváltoztatását csak az általunk — szintén az osztályleírás keretében — megadott módon engedélyezzük. Ezzel meg tudjuk teremteni azt az interfészfelületet, amely azt definiálja, hogy a külvilág objektumai hogyan hathatnak az adott osztály objektumaira, más szempontból nézve viszont nagyon hatékony mód arra, hogy megtilthassuk, hogy bizonyos információkat illetéktelen módon meg lehessen szerezni, illetve hogy bizonyos tevékenységeket illetéktelen módon el lehessen végeztetni akár szándékosan, akár — például egy hiba következményeképp — véletlenül.

Az osztály leírása továbbá tartalmazza azokat az üzeneteket is, amelyeket az osztály objektumai küldhetnek a külvilág más objektumainak.

Itt rejlik a másik fontos különbség a hagyományos és az objektum-orientált programozási módszer között. Míg a hagyományos programozási mód szerint függvényeket hívunk meg adott paraméterekkel, addig az objektum-orientált programozási módszerben az objektumok üzeneteket küldenek egymásnak, és a „fogadó" objektum dönti el, hogyan reagál a vett üzenetre, azaz annak megfelelően milyen tevékenységet végez el. Nagyon gyakran az, hogy az üzenet vételekor melyik konkrét (fizikai) programrészt kell lefuttatni, a fordítás során nem dönthető el, ennek meghatározása csak futási időben történhet meg. Ez az úgynevezett dinamikus kötés. Más szavakkal azt mondhatjuk, hogy az objektum „maga keresi meg" az üzenethez tartozó válasz programkódját.

Tehát láthatjuk, hogy egy osztály (amelyet — kicsit sántító hasonlattal, de — az „adattípus" általánosításának is felfoghatunk) leírása tartalmazza az összes olyan információt, mely az osztály objektumainak állapotát és viselkedését meghatározza. Ez az információ passzív és aktív elemeket egyaránt magában foglal, követve a valóságos modell elemeinek szerkezetét.

A modell egységeinek (az osztályoknak) a definiálása mellett mód van az ezen egységek közötti kapcsolatok meghatározására, definiálására is.

Erre szolgál az objektum-orientált programozási módszer másik fő jellemzője, az öröklődés. Ennek a mechanizmusa teljesen hasonló a modellalkotásnál leírtakhoz. Vagyis lehetőség van arra, hogy egy osztályt úgy definiáljunk, hogy megmondjuk, melyik másik osztályt tekintjük az alapjának (szülő osztály), és megmondjuk, hogy ehhez képest milyen más, speciális tulajdonságokkal rendelkezik.

Az egyszeres öröklődés (amikor a leszármazott osztálynak csak egy szülője van) felel meg a specializációnak — vagy a másik irányból nézve az absztrakciónak —, míg a többszörös öröklődés (amikor a leszármazott osztálynak több szülője van) az aggregáció — illetve a másik irányból nézve a specializáció — természetes megfelelője.

Mivel az öröklődés tetszőleges mélységű lehet, és az egyszeres illetve a többszörös öröklődés is tetszőlegesen kombinálható, tetszőleges bonyolultságú öröklődési struktúra (osztályhierarchia) írható le, amely felépítésében és szerkezetében hűen követi a fizikai modell felépítését, szerkezetét.

Az objektum-orientált programozási elvek szerint elkészített program szerkezete is nagymértékben eltér a hagyományos programszerkezettől. Míg a hagyományos programok nagyrészt adattípusok és függvények definíciójából, valamint az e függvényeket valamilyen sorrendben meghívó főprogramból állnak, addig egy objektum-orientált nyelven megírt program jórészt osztálydefiníciókat tartalmaz egy nagyon rövid főprogram kíséretében, mely főprogram majdhogynem csak az első elküldendő üzenetet tartalmazza. A program további futását az objektumok állapotai és az ezeknek megfelelő kölcsönhatások sorozata szabja meg.

Az objektum-orientált programozási elvek hatása a távközlési szoftverekre

Végezetül nézzük meg, hogy az objektum-orientált programozási elvek használata hogyan segíti elő a távközlési szoftverekkel kapcsolatos elvárások teljesülését!

Mivel az objektum-orientált nyelven készült programok szerkezete, felépítése majdnem teljesen megfelel a megoldandó problémáról alkotott fizikai modellének, ezért az ilyen nyelven írt programok elkészítése egyszerűbb a hagyományos nyelveken készült programokénál. Ez megmutatkozik a programfejlesztés idejének, valamint — közvetett módon — a hibák számának csökkenésében. Mindkét tényező egyben a kifejlesztési költség csökkenésének irányába hat. A program világos szerkezete a fenntartást is megkönnyíti.

Az öröklődés intenzív kihasználása nagyon jelentős hatást gyakorol a forrásprogram-méret csökkenésére. Mivel kevesebb az „írásmennyiség", ez jelentősen lerövidíti a kifejlesztés idejét, csökkenti a hibahelyek számát és ezek által a kifejlesztés költségét.

Mivel az osztályok leírása tartalmazza az osztályra vonatkozó összes jellemzőt, ezért a program átalakításakor, módosításakor csak EGY helyen kell hozzányúlni a már meglévő programkódhoz. A program olyan jellegű továbbfejlesztése pedig, amikor új tulajdonságokat veszünk hozzá a már meglévőkhöz — a specializációt támogató öröklődési mód kihasználásával — nagyon gyakran (új leszármazott osztályok definiálásával) pusztán additív úton, tehát a már meglévő programkód változatlanul hagyásával elvégezhető, mint arra a [2]-ben láthatunk példát.

A programrészek közötti világos, meg nem kerülhető interfészfelületek kialakításának támogatása megkönnyíti a több fejlesztő által készülő, nagyméretű szoftverek előállítását. A precíz interfészfelületek, valamint az adatelrejtés lehetősége pedig korlátozzák a hibák hatásának terjedését — tehát egy esetlegesen a programban maradó hiba nem teszi tönkre a program más részeinek működését —, illetve megkönnyítik a hiba helyének lokalizálását.

Az objektum-orientált programozás alkalmazása további jelentős kényelmi lehetőségeket nyújt a programozónak. Ezek ismertetése meghaladja cikkünk kereteit, az érdeklődők a [2], [3] és [4] irodalomban olvashatnak ezekről részletesebben.

Az objektum-orientált programozás előnyeiért általában a futási sebesség csökkenésével és a fordítóprogram bonyolultságának és méretének növekedésével kell megfizetnünk.

Ezek közül a futási sebesség csökkenése a lényegesebb. Ezt a csökkenést döntő mértékben a dinamikus kötés okozza, vagyis az, hogy a megfelelő programkód megkeresése csak futási időben történik meg. Ez a futási idő csökkenés sok objektum-orientált nyelv esetében annyira számottevő, hogy az ilyen nyelven írt programok valós idejű felhasználásra nem vagy csak korlátozottan alkalmasak.

Azonban van egy olyan, széles körben elterjedt objektum-orientált programozási nyelv, amely ezt a hátrányt jelentős mértékben csökkenti. Ez a C++ nyelv, mely a programozók által jól ismert C nyelv objektum-orientált változata, azzal felülről teljesen kompatibilis.

A C++-ban a dinamikus kötés általi sebességcsökkenést a végrehajtható kód méretének csekély növekedése (további táblázatok beiktatása — lásd részletesen [4]-ben) árán sikerült nagyon leszorítani. A C++ programok átlagos sebessége csak mintegy 30—40%-kal marad el (az amúgy is gyors) C programok sebességétől. Ezzel a C++ nyelv a sebesség terén is felveszi a versenyt. A C++ programok sebessége tovább növelhető a kritikus helyeken a C assembly-közeli lehetőségeinek fokozott kihasználásával, illetve a nyelv elemeivé tett „makrók" (inline függvények

— [2], [4]) használatával, a végrehajtható kód méretének a terhére történő kompromisszumos megoldásként.

3. ÖSSZEFOGLALÁS

A távközlési szoftverek speciális követelményeket állítanak a programozók és a programnyelvek elé. A jelenleg széles körben elterjedt programozási nyelvek ezeket az igényeket csak korlátozott mértékben képesek kielégíteni.

Ezért nagy jelentőségű az új, a programfejlesztést megkönnyítő nyelvgeneráció, az objektum-orientált programozási nyelvek megjelenése.

A fizikai modellalkotás elvét követő, számos, hatékonyan felhasználható jellemzővel rendelkező nyelvek — illetve ezek közül is a rohamosan terjedő, jó sebességi mutatókkal rendelkező C++ nyelv — kiváló eszközöknek bizonyultak a távközlési szoftverek fejlesztése területén.

IRODALOM

[1] Madsen, O.L., Møller-Pedersen, P., "What object-oriented programming may be — and what it does not have to be" *in S. Gjessing—K. Nygaard (ed.): ECOOP'88 European Conference on Object-Oriented programming*, Springer-Verlag, 1988. pp.1—20.

[2] Adamis G., „Távközlési szoftverek és az objektum-orientált programozás", *Magyar Távközlés*, 1992. május pp.6—14.

[3] Stroustrup, B., "What is Object-Oriented Programming?", *IEEE Software*, 1988. május pp.10—20.

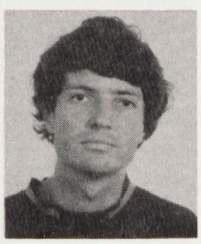
[4] Ellis—Stroustrup, "The Annotated *C++*", Addison-Wesley, 1990.

APPLYING THE OBJECT-ORIENTED PROGRAMMING FOR DEVELOPING TELECOMMUNICATIONS SOFTWARE

G. ADAMIS

DEPARTMENT OF TELECOMMUNICATIONS AND TELEMATICS
TECHNICAL UNIVERSITY OF BUDAPEST
H-1111 BUDAPEST, STOCZEK U 2.

The article describes the impact of the object-oriented programming on the telecommunications software. It presents the special demands of the telecommunications software, describes the most important characteristics of the object-oriented programming (OOP), and the way how the process of modelling is reflected in the principles of the OOP. Finally it summarizes the benefits of the OOP (and especially the C++ language) on creation of telecommunications software.



Adamis Gusztáv 1989-ben szerzett diplomát a Budapesti Műszaki Egyetem Villamosmérnöki Karán. 1989-ben részt vett az egyetemi Végzős Konferencián.
Jelenleg tanszéki mérnök a Budapesti Műszaki Egyetem Távközlési és Telematikai Tanszékén (TTT). Kutatási területe az objektum-orientált programozás és a távközlési protokollok. Részt vesz a BME—TTT és a PKI—TI közös, a No. 7-es távközlési protokollal kapcsolatos projektjében.

SZOFTVERFEJLESZTÉS ELOSZTOTT RENDSZEREK SZÁMÁRA

VALLÓ FERENC

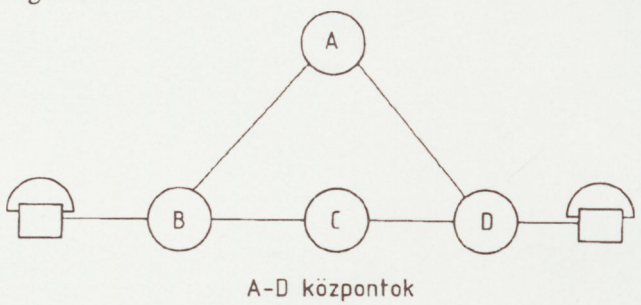
TÁVKÖZLÉSI ÉS TELEMATIKAI TANSZÉK
BUDAPESTI MŰSZAKI EGYETEM
H-1111 BUDAPEST, STOCZEK U. 2.

A cikk néhány problémán keresztül bevezeti az elosztott távközlési rendszer fogalmát. Ezután bemutatja modellezésük matematikai alapját és az ezekre épülő leíró nyelveket. Végül vázolja az említett problémák megoldásának módozatait.

1. BEVEZETÉS

Az elosztott rendszer (distributed system) fogalma széles körben használt, ennek ellenére nem létezik pontos definíciója. Jelen cikkben elosztott rendszernek olyan számítástechnikai rendszert nevezünk, amelyben az erőforrások nem koncentráltan egy egységben helyezkednek el, és közöttük jelentős információcsere zajlik. Ez az információcsere lassúbb és kevésbé megbízható, mint az egyes egységeken belüli kommunikáció.

Elosztott rendszerre jó példa egy telefonhálózat (1. ábra). Ebben az esetben a központok képviselik a számítási kapacitást. E központok egymással kommunikálva hajtanak végre valamilyen tevékenységet. Ez például tipikusan a hívó és hívott fél összekapcsolása, köztük a beszédút biztosítása, majd az összeköttetés végeztével az összeköttetés megszüntetése.



1. ábra. Telefonhálózat mint elosztott rendszer

Jelenleg a telefonközpontok már többnyire digitálisak, így az elosztott rendszer sajátosságai a szoftverben is megjelennek.

A feladat, hogy egy ilyen elosztott rendszer számára fejlesszünk szoftvert. Egyéb (nem elosztott) rendszerekben a szoftverfejlesztésnek jól ismert módszerei vannak. Alkalmazhatók-e ezek a módszerek változtatás nélkül? A válasz: nem. Ugyanis az egymással párhuzamosan működő rendszerek között olyan szinkronizációs és kommunikációs problémák adódnak, amely csak ezen rendszerek sajátja. E problémák speciális megoldásokat kívánnak. A következőkben néhány, csak az elosztott rendszerekre jellemző problémát mutatunk be.

2. NÉHÁNY PROBLÉMA

Sok multiprocesszoros rendszerben a kommunikáció eszköze egy közös memória használata. Tehát a küldő processzor ugyanabba a memóriába ír, amelyből a vevő processzor olvas. Nézzünk meg most két programrészletet, amelyek külön-külön ugyanazt eredményezik:

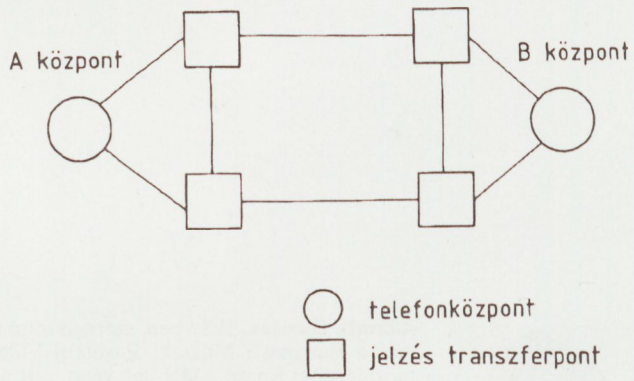
X: = 1;	X: = 0;
	X: = X + 1;
A processzor	*B* processzor

Mindkét processzorban a műveletsor végrehajtása után, az X változóban 1 lesz a tárolt érték. Azonban, ha a két processzor ugyanazon a memóriaterületen dolgozik és a szinkronizáció nincs megoldva, előállhat a következő eset:

1.		X: = 0;
2.	X: = 1 ;	
3.		X: = X + 1;
A processzor		*B* processzor

Mivel X most mindkét processzor számára ugyanaz a memóriadarab, a művelet eredménye X = 2 lesz !!!!

A másik példában tekintsünk egy 7-es jelzésrendszeri hálózatot (2. ábra).



2. ábra. 7-es jelzésrendszeri hálózat

A hálózat célja, hogy az A és B telefonközpont között (a hívásokra vonatkozó) jelzési információkat vigyen át.
A problémák:

- Milyen úton haladjanak az üzenetek?
- Hogyan kerüljük el az üzenetismétlést vagy -vesztést?

Ha ezek mégis előfordulnak, hogyan biztosítsuk a központok helyes működését?

Ezeket a problémákat hívjuk kommunikációs problémáknak.

3. A PROBLÉMÁK KEZELÉSE

Mint általában a műszaki tudományokban, jelen esetben is a legfontosabb a helyes modellalkotás. Az elosztott rendszerek esetében a modellalkotásnak két szintje van: a matematikai modellek szintje és az ezeken alapuló specifikációs és leíró nyelvek szintje.

A matematikai modellek két fő csoportra oszthatók. Az egyik csoportba az ún. processzalgebrák (CSP, CCS) és a kiterjesztett véges automaták (EFSM) tartoznak, a másikba az absztrakt adattípusok (ACT ONE).

A CSP-t (Communicating Sequential Processes) Hoare ismertette 1978-ban [1]. Ez volt az első igazán széles körben elterjedt módszer. A CSP a következő fogalmakat használja a modellalkotásban:

- Őrzött parancs: Ez a szerkezet szekvenciális parancssorozatok megadására szolgál. Minden parancs két részből áll, egy őrből (logikai feltétel) és egy parancsból. Egy parancssorozatból az a parancs hajtódik végre, amelyik őre igaz. Ha több ilyen van, akkor csak egy parancs hajtódik végre. Például az X és az Y változó értékei közül t vegye fel a nagyobbikét, ez CSP-ben:

```
[ x <= y → t: = y,  x > y →  t: =y ]
őr          parancs  őr        parancs
```

- A párhuzamosságot parallel parancsokkal lehet kifejezni. Ezek a parancsok egymással párhuzamosan futnak:

```
[ a : parancs || b : parancs || c : parancs ]
```

- A processzek egymás közötti kommunikációjára input–output parancsok szolgálnak:

a-processz ? változó 1
b-processz ! változó 2

Az *a*-processztől beolvasok valamit a változó 1-be és a *b*-processznek elküldöm a változó 2 tartalmát. Ez a jelölés (? – input, ! – output) később széleskörűen elterjedt. Például az OCCAM nyelvben is használják, amely speciális többprocesszoros rendszerek (ún. transzputerek) programozási nyelve.

Természetesen ezen utasítások szemantikája (jelentése) matematikailag jól definiált.

A másik jól ismert processzalgebrát a CCS (Communicating Concurent Systems) Miller dolgozta ki 1980-ban [2]. Erről bővebb ismertetés Tarnay Katalin cikkében található [3].

Az EFSM (Extended Finite State Machine) nem kapcsolható szorosan egy emberhez vagy dátumhoz. Erről is a [3]-ban olvashatunk részletesebben.

Az absztrakt adattípusok (ADT-k) az előfutárai, illetve megalapozói az objektum-orientált módszereknek. Az ACT ONE az ADT-k egy jó példája. Ezt azonban ritkán használják önállóan, inkább az SDL és LOTOS nyelvek részeként szerepel.

Az ADT-k lényege, hogy a rendszereket mint adatelemek összeségét tekintik. Ezek az adatelemek jól meghatározott struktúrával és műveletekkel rendelkeznek. Jó példa a logikai (boolean) adattípus:

```
newtype Boolean
literals true, false;
operators
        not: Boolean -> Boolean;
axioms
        not(true) == false;
        not(false) == true
endnewtype Boolean;
```

A Boolean típus két értéket vehet fel (true, false). Egy műveletet lehet — jelen példában — végrehajtani rajta, ez a negáció. Ez a művelet egy logikai (Boolean) értéket szolgáltat és axiómák segítségével a művelet is definiálva van. Természetesen ennél sokkal bonyolultabb adatszerkezetek (sorok, tömbök, stb.) is leírhatók ilyen módon.

Megjegyzendő, hogy az ADT-k nem elsősorban elosztott rendszerek modellezésére szolgálnak, azonban alapját képezik az objektum-orientált módszereknek, ezért ismertettük őket.

A matematikai módszerek után lássuk az ezeken alapuló nyelveket. Az SDL és az ESTELLE az EFSM-en alapul, a LOTOS a CCS-en. Ezekről a nyelvekről bővebben a [3]-ban olvashatunk.

Az objektum-orientált módszerek közül a MONDEL az egyik legismertebb. Ezt a nyelvet Bochmann [4] és társai dolgozták ki. A MONDEL egy szigorúan objektum-orientált nyelv, azaz a nyelv minden konstrukciója objektum (szemben például a C^{++} -szal). Így az objektumok az alapegységei a párhuzamosságnak és az elosztottságnak is. A kommunikáció az objektumok között aszinkron módon, eljáráshívással történik. A MONDEL ezenkívül az elosztott rendszerek tervezésének minden fázisát támogatja, a tervezéstől a megvalósításon át a tesztelésig.

4. KÖVETKEZTETÉSEK

Mint láttuk, lehetőség van az elosztott rendszerek modellezésére. E modellek alapján a rendszerek már ellenőrizhetők, illetve tervezésük kézben tartható.

Milyen segítséget adnak e modellek a cikk elején bemutatott problémák megoldására? Az első (szinkronizációs) problémára megoldás, hogy a módszerek jól definiált input–output eljárásokat alkalmaznak, így ezek jól követhetők. A második esetben (kommunikációs probléma) a módszereink segítségével protokollt, azaz egy olyan kommunikációs eljárást kell terveznünk, amely megoldja az említett problémákat. Így az elkészült protokoll könnyen tesztelhető és megvalósítható lesz. A protokolltechnológiáról a [3] cikkben részletesebb ismertetés található.

Látható, hogy a formális módszerek alkalmazására nagy szükség van, ezek teszik lehetővé nagy és komplex rendszerek tervezését és magvalósítását. Mivel ilyen rendszerek tipikusak a távközlésben, a BME Távközlési és Telematikai Tanszékén is folynak ilyen irányú kutatások.

IRODALOM

[1] Hoare, C.A.R., "Communicating Sequential Processes", *Comm. ACM*, Vol. 21, No. 8, pp.666–677 Aug. 1978.

[2] Miller, R.: *Communicating and Concurrency*, Prentice Hall, 1989.

[3] Tarnay K., "Protokollok specifikálása és tesztelése", *Híradástechnika* jelen száma.

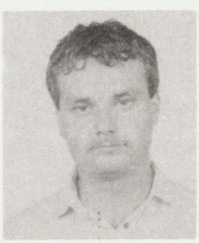
[4] Bochmann, G., Barboeu, M., Erradi, M., Leconte, L., Mondain-Monval, P., Williams, N., "MONDEL: An Object Oriented Specification Language", *Centre de recherche informatique de Montréal*

SOFTWARE DEVELOPMENT FOR DISTRIBUTED SYSTEMS

F. VALLÓ

DEPARTMENT OF TELECOMMUNICATIONS AND TELEMATICS
TECHNICAL UNIVERSITY OF BUDAPEST
H–1111 BUDAPEST, STOCZEK U 2.

The paper introduces the notion of distributed communication systems. Then it deals with the mathematical foundation of modelling the distributed systems and the Formal Discription Techniques (FDTs) based upon them. Finally the author suggests processes for dealing with such systems.



Valló Ferenc 1991-ben végzett a Budapesti Műszaki Egyetem Villamosmérnöki Kar Informatikai ágazatán. Szakterülete a távközlési protokollok és technológiájuk. Már publikált a Magyar Távközlés című folyóiratban. Jelenleg doktori ösztöndíjas a BME Távközlési és Telematikai Tanszékén.

A MAGYAR TÁVKÖZLÉST ÉRINTŐ NEMZETKÖZI TEVÉKENYSÉGRŐL

A magyar politika törekvése, hogy hazánk közeledjen Nyugat-Európához, s hogy az élet számos területén képesek legyünk integrálódni az európai környezetbe. Ez a törekvés két irányban is nagyon komoly feladatokat jelöl ki a távközlés területén és a távközléssel foglalkozók számára. Egyrészt megköveteli a meglévő szolgáltatások nyugat-európai szintű nyújtását, valamint kiterjedt nemzetközi távközlési hálózatok és újabb szolgáltatások létesítését, másrészt igényli a nemzetközi kapcsolatok kiszélesítését a távközlés területén működő más országokbeli partnerekkel, továbbá a távközlési szervezetekkel folyamatos és hosszútávra érvényes együttműködés kiépítését.

1. ALAPVETŐ VÁLTOZÁSOK

Magyarországon a távközlésben érdekeltek nemzetközi tevékenysége az utóbbi néhány évben több szempontból is különbözött a korábbi évtizedek gyakorlatától. Az ezt előidéző okok zömmel három fő csoportba sorolhatók:

- Európa-szerte változik a távközlési környezet;
- hazánkban jelentős politikai és gazdasági változások történtek;
- a magyar távközlést alapvetően átszervezték.

1.1. Távközlési környezet

Az európai távközlési környezet megváltozása természetesen hatással van a magyar távközlésre is, legfőképpen annak nemzetközi kapcsolataira. Általánosnak tekinthető az a tendencia, hogy szétválik a távközlést szabályzó és szolgáltató funkció. Ez a szervezeti módosulás az európai országok nagy részében már megtörtént, egy részében most van folyamatban. Ezzel együtt erős az a tendencia, hogy megszűnjön a távközlés monopóliuma és a szolgáltatások területén is a kereskedelmi tevékenységekre vonatkozó szabályok érvényesüljenek.

A monopólium oldás mélysége — vagyis annak meghatározása, hogy mely tevékenységek maradjanak monopol körben — még nem egységes Európa-szerte. Az Európai Közösségek kezdeményezésére kiadott Zöld Könyv tartalmaz erre vonatkozóan irányelveket, amelyek érvényesülése egyre általánosabbá válik.

A távközlési környezet megváltozásának logikus következménye a nemzetközi szervezetek átrendeződése is. A közvetlen érdekirányok megváltoztak, átformálódnak az érdekközösségek is. Az országonként egyetlen, monopol jellegű távközlési igazgatás, amely ugyanakkor szolgáltató is volt, más viszonyt alakított ki a hasonló helyzetben lévő külföldi partnereivel, mint amire szükségük van jelenleg a versenyhelyzetben lévő távközlési vállalatoknak.

1.2. Politika

A magyarországi politikai változások következtében a nemzetközi kapcsolatok hangsúlya eltolódott. Csökkent a kelet-európai és növekedett a nyugat-európai országokkal az együttműködés jelentősége. Ez az eltolódás nemcsak a magyar kezdeményezésekben mutatkozott meg, hanem szomszédaink részéről is tapasztalható volt. A kelet-európai országokat összefogó regionális szervezetek (OSzSz, KGST–POTÁB) tevékenysége lelassult, majd megszűnt vagy megszűnőben van és lényegesen kevesebb kezdeményezés történt e relációkban a mi részünkről, és a többi ország részéről is.

Az öt közép- és kelet-európai ország tagja lett az eddig csak Európa többi részét összefogó postai és távközlési egyesületnek, a CEPT-nek. Ugyanezek az országok erős érdeklődést tanúsítanak a többi összeurópai távközlési szervezet — mint pl. az ETSI, az Eutelsat stb. — iránt is. Néhányan már csatlakoztak, s a többiek is nemsokára csatlakozni fognak e szervezetekhez.

A szomszédos országokkal eddig meglévő kétoldalú kapcsolatok folyamata is lelassult. Magyar részről és a partnerek részéről is tapasztalható az érdeklődés fokozódása a vezető nyugat-európai távközlési vállalatok és szervezetek felé.

1.3. Átszerveződés

A korábban a Magyar Posta keretében működő hazai távközlés átszervezése során a távközlési feladatok egyrésze a minisztériumhoz került, a vállalati feladatokat pedig megosztva a Magyar Távközlési Rt. (MATÁV), a Magyar Műsorszóró Vállalat (MMV) és számos kft. látja el.

2. A KÖZELMÚLT VÁLTOZÁSAI

A közelmúltban hazánkban három olyan, egymástól viszonylag független hatás érvényesült, amelyek mindegyike a távközlés területén meglévő nemzetközi kapcsolatok szorosabbra fűzését, kiszélesítését, új kapcsolatok létesítését eredményezte:

- A magyar gazdaság export–import tevékenysége Nyugat-Európa felé egyre bővül, s a mindennapi élet természetes részévé válik a gyors és közvetlen kapcsolattartás. A gazdasági élet legkülönbözőbb területei még a korábbinál is fokozotabban igénylik az egyes távközlési szolgáltatások fejlesztését.
- A déli és keleti szomszédos országokban folyó új államalakulások új kapcsolatrendszereket tesznek szükségessé. Teljesen új távközlési hálózatigények jelentkeznek és új irányok nagyon gyors kiépítését követelik.
- Hazánk sok területen igyekszik a regionális csoportosulások fókuszába kerülni és többek között a távközlésben is olyan helyzetben van, hogy célszerű együttműködnie a régióban lévő partnerországokkal, hiszen komoly szerepvállalásra van lehetősége a közép- és kelet-európai csoportosulásokban.

3. NEMZETKÖZI KÖRNYEZET

A távközlés iránti érdeklődés olyan erős, hogy a legkülönbözőbb nemzetközi testületek, szervezetek igyekeznek hatást gyakorolni a fejlesztésére, igyekeznek szabályozni a működését. Jóllehet ezek a hatásmechanizmusok nagyon összetettek, a kép egyszerűsítése érdekében érdemes négy rétegbe sorolni őket:

- a kormányzati,
- az igazgatási,
- a vállalati és
- a felhasználói szférára.

3.1. Kormányzat

A legmagasabb szintet képviselő EK (Európai Közösségek) nagy figyelemmel kíséri a távközlés fejlesztését, több alkalommal közvetlenül is beleszólt, s várhatóan a jövőben is bele fog szólni a távközlési környezet alakításába. Az EK szorgalmazta a távközlési Zöld Könyv elkészítését, amelynek hatására alapvető változások történtek sok országban.

Az EK rendszeresen megrendezi az EK távközlési miniszterek találkozóját, és az ott előkészített témákat az Európai Parlament vagy az EK Tanácsa elé terjeszti. Támogat regionális távközlési projekteket, s a távközléssel foglalkozó EK XIII. Vezérigazgatóság részletesen figyeli a távközlési tevékenységeket nemcsak az EK tagországokban, hanem pl. hazánkban is.

Jóllehet az EK határozatok Magyarországra közvetlenül még nem vonatkoznak, de a nemrégen aláírt Társulási Szerződés már jelzi, hogy a nem túl távoli jövőben már nekünk is kötelezően figyelembe kell venni az elfogadott irányelveket, határozatokat.

3.2. Igazgatás

A távközlési igazgatásokat a Nemzetközi Távközlési Egyesület (ITU) fogja össze. A közeljövőben jelentősen át fogják alakítani, megváltozik a szervezete és a munkamódszere is. Az európai regionális szervezet, a CEPT most van átalakulóban. Ez évben meg fognak szűnni bizonyos tevékenységek és az igazgatások részvétele is sok szempontból más formájú lesz.

3.3. Vállalat

A vállalati szféra nemzetközi szervezetei most vannak kialakulóban. Nyilvánvaló, hogy az egyes távközlési vállalatok munkája összehangolt tevékenységet igényel annak érdekében, hogy minél több szolgáltatás legyen nemzetközileg is elterjeszthető. Ugyanekkor ezek a vállalatok egymással versenyben állnak, igyekeznek egymást kiszorítani a távközlés-szolgáltatási piacokról. Ezt az ellentmondást csak komoly munkával lehet kiküszöbölni oly módon, hogy minden egyes témánál külön-külön meg kell vizsgálni a kooperáció és konkurencia összeegyeztethetőségét.

A vállalati síkhoz tartozó legfontosabb nemzetközi szervezet az európai hálózatüzemeltetőket egyesítő ETNO, amelynek létrehozásában a MATÁV is résztvett. Az ETNO keretében kifejezetten a tradicionális távközlési szolgáltató vállalatok tömörülnek, főként azért, hogy összehangolják a hálózat- és szolgáltatásfejlesztési terveiket, és olyan közös álláspontot alakítsanak ki a valamennyiüket érintő témákban, amelyet a távközlési terület más résztvevői felé közösen érvényesíthetnek.

3.4. Felhasználók

Nyugat-Európában egyre jelentősebb szerephez jutnak a felhasználók egyesületei. Jelentkeznek igényeikkel és követelményeket támasztanak a szolgáltatásokkal szemben. A kezdeti bizonytalanságok után az egymással versenyben álló távközlési vállalatok egyre inkább felismerik, hogy megfelelő együttműködés kialakítása esetén értékes partnerekre találhatnak bennük, s érdemes komolyan figyelembe venni véleményüket.

4. NEMZETKÖZI SZERVEZETEK

A figyelem előterében lévő nemzetközi szervezetekről a következő áttekintés adható:

4.1. ITU (International Telecommunication Union)

A Nemzetközi Távközlési Egyesület jogelődje 1865-ben alakult meg Párizsban. Az Egyesület 1948-tól az ENSZ szakosított szervezeteként működik, jelenleg tagja a világ 164 országa.

Az Egyesület célja a nemzetközi együttműködés kiterjesztése a távközlési szolgálatok hatékony működtetése és fejlesztése, valamint a szolgáltatások bővítése terén. Feladatait a következő szervezeti felépítésben látja el:

- Meghatalmazottak Értekezlete
- Igazgatási Értekezlet
- Igazgatási Tanács
- Főtitkárság
- Nemzetközi Frekvencianyilvántartó Bizottság – IFRB
- Nemzetközi Rádió Tanácsadó Bizottság – CCIR
- Nemzetközi Távíró és Távbeszélő Tanácsadó Bizottság – CCITT
- Távközlésfejlesztési Iroda – BDT

A Meghatalmazottak Értekezlete, a szervezet legfőbb törvényalkotó fóruma határozatokat fogad el, irányítja az Egyesületet, együttműködési megállapodásokat köt más nemzetközi szervezetekkel, meghatározza az Egyesület célkitűzéseit.

Az Igazgatási Világértekezlet valamely sajátos távközlési ág szabályzatait vizsgálja felül vagy az egész világot érintő kérdéseket vitat meg. Legismertebbek a Rádióigazgatási Világértekezletek (WARC), amelyek általában egy-egy frekvenciasáv használati és területi felosztásával foglalkoznak.

Az Igazgatási Tanács dönt olyan kérdésekben, amelyek meghaladják a főtitkár hatáskörét, de az Egyesület folyamatos működtetéséhez szükségesek. Feladatait és kompetenciáját az Egyesület alapokmányai írják elő. Tagjait a Meghatalmazottak Értekezlete választja meg.

A Főtitkárság élén a főtitkár, illetve annak helyettese áll. A Főtitkárság végzi az Egyesület napi adminisztratív munkáit és ellátja az értekezletek titkársági teendőit is.

A Nemzetközi Távíró és Távbeszélő Tanácsadó Bizottság (CCITT) feladata, hogy tanulmányokat végezzen és ajánlásokat dolgozzon ki a távközlési szolgálatok műszaki, kezelési és díjazási kérdéseiben. Egy-egy CCITT ajánlás komoly kollektív munka eredménye, s habár nincs kötelező jellege, mégis a gyártó ipar nagy figyelmet szentel ezekre és a távközlési projektek kialakításánál meghatározó szerepük van.

A Nemzetközi Rádió Tanácsadó Bizottság (CCIR) tanulmányokat folytat és ajánlásokat dolgoz ki a rádiótávköz-

lés területén. A CCIR-ben jelenleg 10 tanulmányi bizottság működik, s a négyévenként összeülő Közgyűlésen határozzák meg a tanulmányozandó kérdések körét és fogadják el, illetve aktualizálják az ajánlásokat.

A Nemzetközi Frekvencianyilvántartó Bizottság (IFRB) az egyes országokban használt, nemzetközi védelemre igényt tartó vagy zavarást okozni képes frekvenciafelhasználásokat ellenőrzi és regisztrálja. A nyilvántartás elektronikus eszközökkel, nagyszámítógépekkel történik. A frekvenciákkal kapcsolatos tevékenység mellett az IFRB részt vesz a Rádióigazgatási Világértekezletek (WARC) előkészítésében.

A Távközlésfejlesztési Iroda (BDT) célja a fejlődő országok részére műszaki segítség intézményesítése a távközlésfejlesztés területén. A nizzai Maghatalmazotti Értekezlet teremtette meg létének és működésének jogi kereteit 1989-ben, a szervezet jelenleg alakul.

A legutóbbi Meghatalmazotti Értekezlet Magasszintű Bizottságot (HLC) hozott létre az ITU működésének, szervezeti felépítésének felülvizsgálatára. Ez a Bizottság jelentést készített, amelyben 96 ajánlást tett az ITU szerkezetének módosítására. Ezeket az ajánlásokat 1992 végén kifejezetten e célra összehívott Meghatalmazotti Értekezlet fogja megvizsgálni. Várhatóan ez az értekezlet jelentős változtatásokat fog bevezetni annak érdekében, hogy az ITU jól alkalmazkodjon a megváltozott távközlési környezethez, továbbá növelje a világméretű távközlési koordinációra irányuló munkájának hatékonyságát.

4.2. CEPT (Conference Europenne des Administrations des Postes et des Télécommunications)

A szervezetet 1959-ben 19 nyugat-európai ország hozta létre, később további európai államok is csatlakoztak. Jelenleg a volt Szovjetunió kivételével valamennyi európai ország, összesen 32 állam postai és távközlési szervezeteit tömöríti. Hazánk 1990-ben csatlakozott a CEPT-hez.

A CEPT legfontosabb célja, hogy tagjai számára európai szintű egyeztetési lehetőséget biztosítson feladataik összehangolására.

A CEPT több évtizedes munkája során, a nyugat-európai fejlett távközléssel rendelkező országok szakembereinek összefogása révén, nagy számban szerzett tapasztalatokat, sok ajánlást dolgozott ki, a szervezet keretében többoldalú szerződések jöttek létre egy-egy pontosan megfogalmazott cél realizálása érdekében. E tapasztalatok átvétele nagyon jelentős lehet a magyar távközlési szakemberek számára is.

Az európai távközlési környezet jelentős megváltozása a CEPT-ben is megkíván szervezeti és működési átalakításokat.

Napjainkban, ennek a cikknek a megjelenése idején folynak azok a tárgyalások, amelyek dönteni fognak a CEPT átalakításáról.

Az egyes országok távközlési szervezeteinek közvetlen intézkedését vagy kötelezettségvállalását igénylő témákban sokoldalú Egyetértési Okmányokat (Memorandum of Understanding, MOU) készítettek elő, hogy azokat az érdekelt vállalatok és igazgatások aláírják. Így azok a témák, amelyek többségükben projektek megvalósítására és új szolgáltatások bevezetésére vonatkoznak, a CEPT-ből átkerültek a kompetens MOU-k keretébe. A legfontosabb MOU-k a következők:

- METRAN. A METRAN MOU olyan európai átviteli hálózat létrehozását irányozza elő, amely hordozó szolgálatot biztosít a távközlési hálózatüzemeltetők részére. Felöleli a digitális átviteli képességek, a multiplex és cross-connect funkciók összeségét. A MATÁV csatlakozott a MUO-hoz, s az Irányító Testület munkájába is bekapcsolódott.
- OSS (One–Stop–Shopping — minden szolgáltatást egy helyen). A távközlési szolgáltatások rendelkezésre bocsátásának, a kapcsolatfelvételi pontok közötti együttműködésnek, az adatok kicserélésének és élőn tartásának automatizált rendszererére vonatkozó MOU. Bármilyen bonyolult és kiterjedt nemzetközi szolgáltatást igénybe venni szándékozó ügyfélnek nem kell külön-külön megállapodást kötnie valamennyi országban, hanem az OSS-ben részt vevő bármelyik szolgáltatóval megállapodhat, ami érvényes a többi partnerre is. Az OSS-alapelvek szerint előkészítés alatt vannak kétoldalú megállapodások a MATÁV és a vezető európai távközlési szolgáltató vállalatok között (British Telecom, PTT Telecom Netherlands stb.).
- ISDN. Az 1989-ben aláírt Egyetértési Okmány megállapodás az ISDN szolgáltatások bevezetéséről, egyeztetett nemzetközi interfészek alkalmazásával és a minimálisan kötelező szolgáltatások felsorolásával. Az egyezmény volt az ETSI szabványosítási munkáinak egyik hajtómotorja. A MATÁV szakértői most vizsgálják a lehetőségeket és a csatlakozási feltételeket.
- GSM. A GSM a 900 MHz-es pán-európai digitális kiscellás rádiótelefon-hálózat létrehozását irányozza elő, egyeztetett nemzetközi előírásokra támaszkodva és több országban egyidejűleg. Sajnos az embargós tiltások következtében Magyarország számára korábban nem volt lehetőség a csatlakozásra, de ezek feloldását követően tervezett a magyar részvétel.
- EURESCOM. Az európai távközlés-fejlesztési és kutatási tevékenység összefogására, nemzetközi munkakapcsolatok létesítésére, s a szellemi potenciál integrálására alapított intézmény, mely részvénytársasági formában működik. A MATÁV belépésének előkészítése folyamatban van.

Az említetteken kívül még más MOU-k is vannak, amelyek egy-egy szolgáltatás nemzetközileg összehangolt bevezetésére vagy adott témákban közös fejlesztésekre irányulnak.

4.3. ETNO Alapító Csoport

Az európai távközlési környezet gyors változtatása miatt a CEPT-együttműködés alapvetően átrendeződik. Az EK (Európai Közösségek) igen erős hatást gyakorol a távközlési monopóliumok megszűntetésére és a versenyszféra kiterjesztésére. A postai és távközlési szolgáltató vállalatok szétválnak, elkülönülnek egymástól a szabályozó hatóságok és a szolgáltató vállalatok. A távközlés területén a változások oda vezettek, hogy a nyilvános, állandó helyű és országos távközlési szolgáltatók elhatározták saját szervezetük, az ETNO létrehozását.

Az ETNO körültekintő, valamennyi részletet figyelembe vevő kialakítása érdekében az érdekelt európai vállalatok Alapító Csoportot hoztak létre, amelynek feladatává tették — többek között — az Egyesület Alapokmányának és Általános Belső Szabályzatának kidolgozását. Az Alapító

Csoport egyik alelnökének a MATÁV képviselőjét választották meg.

Az ETNO Alapító Csoport első ülésének résztvevői a MATÁV alapító tagságát és aktivitását annak jeleként értékelték, hogy a nemzetközi politikai fórumokon már korábban bejelentett igyekezet Magyarországon olyan változásokat eredményezett, amelyek az ország Európához való közeledését tényekkel támasztják alá. A magyar alelnök feladatának tekintette, hogy előkészítse az utat a többi közép- és kelet-európai távközlési vállalat csatlakozására is.

4.4. ETNO (Association of European Public Telecommunication Network Operators)

Az ETNO Egyetértési Okmányának (ETNO-MOU) aláírói kifejezték együttműködési érdekeltségüket arra vonatkozóan, hogy létrehozzák az Európai Nyilvános Távközlési Hálózatüzemeltetők Egyesületét. Az okmányt aláíró vállalatok felismerték, hogy mivel — Európa legtöbb országában új távközlési szabályozási környezet jött létre — szükség van a szabályozói és üzemviteli tevékenység egyértelmű különválasztására a nemzetközi szervezetek szintjén is, valamint elengedhetetlen az integrálódás koordinálása az európai hálózatok területén.

Az ETNO az a fórum, ahol az európai nyilvános távközlési hálózatüzemeltetők megvitathatják és egyeztethetik álláspontjaikat, s amely az egységesen kialakított nézeteiket képviselheti más szervezetek felé, különösen az Európai Közösségek Bizottságában.

A MATÁV vezérigazgatója 1991 elején írta alá az ETNO alapítására és az Alapító csoport létrehozására készült Egyetértési Okmányt, majd 1992 májusában az Alakuló Közgyűlésen az Alapokmányt. A második világháborút követően ez volt az első alkalom, hogy magyar vállalat alapító tagja lehetett egy összeurópai távközlési szervezetnek.

4.5. ETSI (European Telecommunication Standard Institute)

A CEPT tagországok a Zöld Könyv ajánlásának megfelelően 1988-ban hozták létre az ETSI-t, az Európai Távközlési Szabvány Intézetet Franciaországban, a Nizza melleti Sophia-Antipolisban.

Az ETSI működésének célja egységes európai távközlési normák kidolgozása oly módon, hogy közösen figyelembe veszik az egyes CEPT tagországok és az azokban lévő különböző csoportok érdekeit. Az ETSI tagjai lehetnek nemzeti szabványosítási szervezetek, távközlési igazgatások, nyilvános hálózati üzemeltetők, kutatóintézetek, iparvállalatok, felhasználói egyesületek stb.

Az ETSI legfontosabb fóruma a Közgyülés, amely általános intézményi irányelveket dolgoz ki, dönt az Alapszabály és a költségvetés kérdésében. A műszaki tevékenységet a Műszaki Közgyűlés tervezi, amely a műszaki bizottságok és projektcsoportok által kidolgozott anyagokat tekinti át. A műszaki bizottságok közös platformot alakítanak ki és munkacsoportokat hoznak létre a szabványok fejlesztésére. A projektcsoportok célirányosan és meghatározott időtartamra szerződnek egy-egy feladat elvégzésére.

Az ETSI gyakorlatilag teljesen átvette a szabványosítási munkát a CEPT-től. Magyarországról a MATÁV a szervezet társult tagja 1991 végétől. Várhatóan rövidesen Magyarország hivatalosan és a megkívánt valamennyi feltételt teljesítve is csatlakozni fog, s ekkor a társult tagság teljes értékű tagsággá válik.

4.6. ETIS (European Telecommunication Informatics Services)

Napjainkban az informatika és a távközlés kölcsönhatása nagyon intenzívvé vált. Egyrészt az informatikai szigetek összekapcsolása komoly távközlési eszközöket igényel, másrészt a távközlés üzemeltetése alapvetően az informatikai eszközök igénybevétele felé tolódik el.

Az európai országokban a távközlési hálózatok vezérlésébe, az üzemvitelbe beépülő informatika fejlesztését nem egységesen, hanem országonként külön-külön kezdték. Néhány éve felismerték, hogy a távközlési szolgáltatók összefogása és közös fellépése az informatikai szoftverkészítők felé egységesebb piacot, jobb működési feltételeket és — végeredményében — olcsóbb szoftverfejlesztést eredményez. Ezért a távközlési üzemeltetők elhatározták, hogy olyan szervezetet hoznak létre, amelynek keretében alkalmuk van megtárgyalni a távközlés területén igényelt szoftverek fejlesztési szempontjait, lehetőséget adva azok egységesítésére. Elhatározták összeurópai távközlési informatikai szervezetként az ETIS megalakítását.

Az ETIS célja a távközlésben alkalmazott szoftverek egységesítése, azok fejlesztési, előállítási és karbantartási költségeinek csökkentése, továbbá a távközlési szolgáltatók közötti ezirányú együttműködés és információcsere elősegítése. Közös elhatározással és az érdekek összehangolásával az ETIS egyeztetett követelményeket dolgoz ki a szoftveralkalmazásokra.

A piaci igények előrejelzésével, a hatalmas volumenből következő tárgyalási pozíció kihasználásával, az ajánlatkérések egységesítésével és egyszerűsítésével az ETIS közvetít a távközlési cégek és a szoftvergyártók között. Ez a tendencia magával vonja majd az egységes európai távközlési informatika létrehozását, az európai hálózatok együttműködésének egyszerűsítését, az egységes szolgáltatásrendszer megvalósítását.

Az ETIS a MATÁV-val együtt 1991 közepén Budapesten tartotta hivatalos alakuló konferenciáját. A MATÁV alapító tagként aktív munkát vállal a szervezet Igazgató Tanácsában.

4.7. INTELSAT (International Telecommunications Satellite Organization)

A Nemzetközi Távközlési Műholdas Szervezetet 11 ország alapította 1964-ben. Napjainkra már 114 ország a szervezet tagja, amely 16 működő műholddal rendelkezik. A Szervezet fő tevékenysége a nemzetközi, különösen az interkontinentális távbeszélő-átvitel, de emellett jelentős az adat- és videojel-továbbítás is. Kisebb volumenű a műholdas műsorszóró szolgáltatás. A Szervezet 1973-ban alakult át kormányközi szervezetté, és akkor dolgozták ki az Egyezményt és az Üzemeltetési Megállapodást is.

A MATÁV rendszeresen igénybe veszi a Szervezet által üzemeltetett áramköröket, s építi a csatlakozó földi állomást. Részt vesz a Szervezet forgalmi és műszaki kérdésekkel foglalkozó ülésein, annak ellenére, hogy hivatalosan még nem tagja az INTELSAT-nak. A magyar igazgatás vizsgálja a Szervezetbe való belépés lehetőségét.

Az INTELSAT a MATÁV-val közösen 1991 közepén szemináriumot szervezett Budapesten, amelynek témakörei között szerepelt az INTELSAT szolgáltatások bemutatása és a VSAT üzemeltetés is.

4.8. INTERSZPUTNYIK

Az INTERSZPUTNYIK Moszkva székhelyű, önálló gazdálkodású, nyílt nemzetközi szervezet. Tagjai a közép- és kelet-európai országok, továbbá több ázsiai és amerikai fejlődő állam. A szervezet célja: az együttműködés és koordináció biztosítása a nemzetközi űrtávközlési rendszer tervezése, fejlesztése és üzemeltetése területén. Az INTERSZPUTNYIK 1982-ben tért át a kereskedelmi alapú szolgáltatás nyújtására. Az utóbbi időben az INTERSZPUTNYIK működési mechanizmusa egy nemzetközi részvénytársasági forma irányába mozdult el.

Az INTERSZPUTNYIK nemzetközi űrtávközlési rendszer részei: az Oroszországtól bérelt két műholdon lévő űrkomplexum és a műholdakkal kapcsolatot tartó földi állomások.

Az INTERSZPUTNYIK szervezet legfelsőbb fóruma az évenként összeülő Tanács, amelyen a tagországok kormányképviselői vesznek részt. A rendszerrel összefüggő üzemviteli, fejlesztési és gazdasági–jogi kérdéseket évenként szakértői értekezleteken tekintik át.

4.9. EUTELSAT (European Telecommunications Satellite Organization)

Az európai Távközlési Műholdas Szervezet olyan kormányközi szervezet, amelynek 28 európai ország a tagja, 6 műholdat üzemeltet, távbeszélő-, adat- és tv-átviteli szolgáltatást nyújt. Tevékenysége Európára korlátozódik. Az EUTELSAT szervezeti felépítése: Közgyűlés, Aláírók Tanácsa, Végrehajtó Szerv, mely utóbbi élén a vezérigazgató áll.

Az EUTELSAT Egyezményhez a kormányok csatlakoznak és általában megjelölnek egy távközlési vállalatot, amely az Üzemeltetési Megállapodás (Operating Agreement) aláírójaként végzi a gyakorlati munkát.

4.10. EEMA (Eurpoean Electronic Messaging Association)

A Szervezetet 1987-ben alapították. Célkitűzése, hogy Európán belül az üzenetközvetítést érintő fejlesztéseket elősegítse és koordinálja a távközlés legkülönbözőbb területein: telex, telefax, teletex, E-mail, EDI, EFT számítógépes konferencia. Céljai között szerepel még a nyilvános nemzetközi szabványokon alapuló interkonnektivitás megteremtése a nyilvános és privát üzenetközvetítő rendszerek és hálózatok között.

Az EEMA 1990 kora őszén Budapesten rendezte meg konferenciáját, amelynek megszervezésében és szponzorálásában a MATÁV is részt vállalt. A konferencia elnöki tisztének betöltésére a MATÁV képviselőjét kérték fel és az EEMA Tanácsadó Testületébe meghívták a vállalat vezetőjét.

5. NEMZETKÖZI CSOPORTOSULÁSOK

A politikai kezdeményezésre létre jött Pentagonálé (Magyarország, Csehszlovákia, Ausztria, Jugoszlávia, Olaszország) előirányzott együttműködést a távközlést illetően is. Az első biztató lépések után a politikai események következtében a lendület megtört. Várhatóan a közös munka folytatása és kibontakoztatása a helyzet normalizálódása és a politikai akarat ismételt kinyilvánítása után lesz lehetséges.

Az 1991. évi visegrádi magas szintű hármas találkozó nyilatkozatában foglaltak szerint mind igazgatási, mind vállalati együttműködés kezdődött a cseh, a szlovák és a lengyel partnerekkel. 1991-ben a MATÁV rendezett vállalati találkozót Magyarországon, amelyen megállapodás jött létre a további együttműködés módjára is. 1992-ben két igazgatási és egy vállalati értekezletre került sor.

6. TÖREKVÉSEK

A távközlési környezet változásai és a magyarországi politikai változások felgyorsították és színesebbé, eredményesebbé tették a nemzetközi munkát. Mind a vezetés, mind a szakembergárda gyorsan és jól alkalmazkodott a megváltozott körülményekhez. Számos kétoldalú kapcsolat létesült, a magyar érdekeltek bekapcsolódtak a nemzetközi szervezetek munkájába, s több irányba tett kezdeményezéseik számottevő hozzájárulásként értékelhetők.

Ezeknek az erőfeszítéseknek köszönhetően a magyar hozzájárulásokat a nemzetközi szakközvélemény már nem csupán szemlélődésnek minősíti, hanem egyenrangú partnerként fogadnak el. Természetesen még sok és komoly munka szükséges ahhoz, hogy a több évtizedes elmaradást és a kirekesztettség következményeit teljesen fel lehessen számolni.

A magyar részvétel elismerését és szakembereink megbecsülését jelzi, hogy a MATÁV nyolc szakértője választott tisztségviselője a különböző nemzetközi szervezeteknek.

Általában jó visszhangra talált, hogy több jelentős nemzetközi konferenciát rendeztek Magyarországon. Ezeken lehetőség nyílott a hazai szakemberek nagyobb számú részvételére és a nemzetközi kapcsolatrendszerhez való közelkerülésre. Ugyanakkor a rendezvények jó alkalmat adtak az európai, néha a tengerentúli partnerek részére is a magyarországi élet és a körülmények megismerésére.

VILLÁNYI OTTÓ
Magyar Távközlési Rt.

KÖNYVISMERTETÉSEK

Géher Károly, Solymosi János
LINEÁRIS ÁRAMKÖRÖK TERVEZÉSE
Tankönyvkiadó, Budapest, 1992.

Jelentős hozzájárulással bővült az áramkörök tervezését megalapozó magyar nyelvű könyvek csoportja. A „Lineáris áramkörök tervezése" című könyv, amely egyetemi tankönyv is, több évtizedes oktató– és kutatómunka eredménye. Gazdag előzménnyel rendelkezik: Géher Károly: „Lineáris hálózatok" és Solymosi János: „Fejezetek a hálózatelméletből" című művének közös, átdolgozott kiadása. Tematikailag kapcsolódik K. Géher: „Theory of Network Tolerances" és Gordos Géza: „Példatár a lineáris hálózatokhoz" című munkájához is.

A könyv a koncentrált paraméterű, időinvariáns, lineáris áramkörök tervezésével kapcsolatos alapvető kérdések teljes körének bemutatására törekszik: a folytonos idejű, passzív hálózatok szintézisének ismertetése mellé bekerült a könyvbe az aktív RC kétkapuk szintézise és a mintavételező hálózatok problémaköre is. A szintézismódszerek összefoglalását az érzékenységekkel és a toleranciákkal foglalkozó fontos fejezet egészíti ki.

A könyv hét fejezetből áll, melyek címei és témakörei a következők:

1. Bevezetés. A hálózatelmélet kialakulásának történeti áttekintése, elméleti alapfogalmak.

2. Passzív kétpólusok szintézise. A megengedett függvények körének meghatározása, LC, RC és GL egykapuk Foster- és Cauer-féle szintézise, minimál impedancia, RLC egykapuk Brune-féle szintézise.

3. Kétkapuk analízise. Kétkapu-paraméterek értelmezése, szimmetria, Barlett–Brune tétel, kétkapuk összekapcsolása, az átviteli tényező megengedett függvényei, Hurwitz-polinomokra vonatkozó kritériumok, az átviteli függvény felbontása minimálfázisú és mindentáteresztő részre, minimálfázisú hálózatok tulajdonságai, Bode formulák, hullámparaméterek, reflexiós mátrix.

4. Érzékenységek és toleranciák. Tolerancia-számítás, tolerancia-központosítás és tolerancia-kiosztás, érzékenységinvariánsok.

5. Passzív kétkapuk szintézise. A megengedett transzfer függvények áttekintése különböző gerjesztések és lezárások esetén, approximáció, megengedett impedanciamátrixok különböző hálózatosztályok esetén, pozitív reális mátrixok, realizálás létrahálózattal, csillapítók, amplitúdó- és futási idő korrektorok, késleltető művonalak és szűrők tervezése.

6. Aktív RC kétkapuk szintézise. Hatásgráfok, direkt szintézis műveleti erősítővel, vezérelt generátorral, negatív impedancia konverterrel, girátorral, LC létrahálózatok szimulációja, kaszkád szintézis.

7. Mintavételező hálózatok (definíció, osztályozás, diszkrét sorozatok Laplace- és z-transzformáltja, hálózatjellemző függvények FIR és IIR szűrők esetén, realizálás, hullámdigitális és kapcsolt kapacitású szűrők.

Az elvi kérdések megértését számos, részletesen kidolgozott példa segíti. Bőséges táblázatanyag növeli a könyv használhatóságát. A könyvet eredményesen használhatják az áramkörtervezéssel foglalkozó villamosmérnökök és villamosmérnök-hallgatók.

LADVÁNSZKY JÁNOS

F. Belina, D. Hogrefe, A. Sarma
SDL WITH APPLICATIONS FROM PROTOCOL SPECIFICATION
(SDL protokoll specifikációs alkalmazásokkal)

A távközlési szoftverek készítését több magas szintű nyelv könnyíti meg, ezek közül egyik legelterjedtebb az SDL (Specification and Description Language). Az SDL nyelvet a CCITT szakértői bizottsága dolgozta ki és a CCITT 1988-as Kék könyvében Z.100-as ajánlásként publikálta. Az „SDL with applications from protocol specification" c. könyv szerzői a Z.100 ajánlás kidolgozói voltak a magyar származású Belina Ferenc vezetésével, így a könyv a szabványosítás eredményét valóban első kézből nyújtja.

A könyv kilenc fejezetből áll és az SDL nyelvet három részre bontva ismerteti. Először a 2. fejezetben áttekinti a nyelv főbb tulajdonságait. A tulajdonságok közül kiemelt jelentőségű a rendszer viselkedése, vagyis hogy mit cselekszik, milyen lépéseket tesz a rendszer adott környezeti feltételek mellett. A rendszer az SDL nyelv legnagyobb egysége és a nyelvi hierarchiában a rendszert a blokk, majd a processz követi.

A szerzők az SDL ismertetését a következő fejezetekben (3–6. fejezet) didaktikusan építik fel, hangsúlyozva a hierarchikus specifikálás előnyeit. E fejezetekben az SDL nyelv két változatát, a grafikus és a programozási nyelvet párhuzamosan mutatják be. A grafikus szimbólumok és a kulcsszavak párhuzamos tárgyalása különösen a mérnök olvasók számára könnyíti meg a nyelv megértését és megtanulását.

A könyv harmadik része (7–9. fejezet) már gyakorlottabb SDL programozóknak szól. Ez a rész az absztrakt adattípusok használatát mutatja be, világosan megfogalmazva és sok egyszerű példával szemléltetve az elméleti alapokat. Ez a rész megismétli mélyebben és részletesebben a nyelv szabályait, bemutatja a nyelv alapját képező véges automatát, a nyelv dinamikus szemantikáját, végül egy igen részletes példával, az INRES protokoll formális specifikálásával szemlélteti az SDL nyelv alkalmazását.

A könyv szerves része a függelék, amely az SDL nyelv szabályait Backus-Naur formában írja le és összefoglalja a grafikus szimbólumokat.

A könyv a Prentice Hall International kiadásában jelent meg angolul, a Carl Hanson Verlag kiadásában németül. A könyv szép kiállítású, jól szerkesztett. Erőssége a sok ábra, ami lényegesen megkönnyíti a megértést.

Kiknek szól a könyv? Elsősorban azoknak, akik meg kívánják tanulni az SDL nyelvet. E célszám szerkesztője és e könyvismertetés szerzője eredményesen használta az „SDL with applications from protocol specification" c. könyvet tankönyvként végzős, informatika szakos hallgatók oktatására a „Kommunikációs rendszerek programozása" c. tárgy keretében. A hallgatók szívesen forgatták a könyv kezükbe adott részleteit és megkedvelték az SDL nyelvet. A könyv másodsorban azoknak szól, akik különböző CCITT ajánlásokat olvasva, az ajánlások SDL nyelvű részleteit és magyarázatait szeretnék jobban megérteni.

A könyv jó és hasznos olvasmány. A hazai távközlési rendszerek növekvő száma miatt érdemes lenne magyar nyelvre is lefordítani.

TARNAY KATALIN

FERENCZY PÁL

1931. SZEPTEMBER 16. – 1992. AUGUSZTUS 12.

Megrendülten értesültünk közvetlen előzmény nélküli, hirtelen haláláról. Nagyon nehéz tudomásul venni, hogy a „Video- és hangrendszerek" előadásainak második félévét már nem tarthatja meg, nem láthatjuk a tv-laborban vagy az íróasztalánál és — hallgatói, munkatársai, barátai, hozzátartozói — nem találkozhatunk többé Ferenczy Pállal.

A Budapesti Műszaki Egyetem hallgatója volt 1950-től, kiváló adottságaival korán kitűnt a társai közül. Utolsó éves hallgatóként mint demonstrátor dolgozott a tanszék televízióval kapcsolatos munkáiban. Akkoriban hazánkban a televízió még csak tantárgyként és távlati tervként létezett. Az akkor vállalt munkája kezdete lett az egész életre kiható elkötelezettségének a hírközlés, és különösen a televízió iránt. Az elmúlt csaknem négy évtizedben ezt az ügyet oktatással, nagyon előretekintő és következetesen megvalósított kutatással, bárkiéhez alig hasonlítható eredményekkel szolgálta.

Oktatói beosztásai: 1954-től tanársegéd volt 1962-ig, majd adjunktus; 1973-ban docenssé nevezték ki és 1980-tól egyetemi tanár. 1980–86 között a Villamosmérnöki Kar tudományos dékánhelyettese volt. A volt Híradástechnikai Elektronika Intézet (HEI) Rádió, Televízió és Hírközlés Osztályának vezetője, majd egy évig a HEI egyik utódaként alapított Híradástechnikai Tanszék vezetője volt 1991–92-ben.

1965-ben lett egyetemi doktor, 1972-ben a műszaki tudomány kandidátusa, majd a műszaki tudomány doktora 1980-ban.

Egyetemi oktatóként számos híradástechnikai tárgyat adott elő, ezek között a Hírközléselmélet, a Televíziótechnika és a Vido- és hangrendszerek című tárgyakat.

Fel kell idéznünk, hogy 1959-ben a híradástechnika szakon bevezetett új szakmai alaptárgyak egyike volt a Hírközléselmélet. A tárgy előadására Ferenczy Pált kérték fel, aki a feladatot kitűnően oldotta meg. Munkája kezdeményező jellegét mutatja, hogy jól megalapozott kötelező tárgyként volt jelen e témakör a hazai képzésben akkor, amikor külföldi neves egyetemeken is csak a felsőbb tanulmányok választható tárgyai között szerepelt.

Tudományos kutatómunkája alapvetően a televíziótechnika témaköréhez kapcsolódik. A hazai tv-adóhálózat kiépítésekor a 60-as évek elején Ferenczy Pál és munkatársai tervezték a kékesi és a kabhegyi tv-adóberendezések közösítő szűrőit (filterplexer).

Ferenczy Pál nagy érdeme, hogy a 70-es évek második felében már a kezdetben felismerte a Teletext (képújság) és a Viewdata nagy jelentőségét és kezdeményezte hazai megvalósítását. Az Országos Műszaki Fejlesztési Bizottság megbízásából Ferenczy Pál és munkatársai tervezték és készítették el azokat a berendezéseket, amelyekkel 1981-ben — nemcsak elsőként a „keleti blokk" országaiban, hanem jónéhány nyugat-európai országnál is korábban — megindult hazánkban a képújság sugárzása. Később ő dolgozta ki a képújság magyar változatának rendszerét és áramköreit, mely szolgáltatja a teljes magyar betű- és ékezetkészletet.

Ferenczy Pál 1960-tól rendszeresen részt vett a Nemzetközi Elektrotechnikai Bizottság (International Electrotechnical Commission, IEC) munkáiban. Szakmai felkészültsége és kiváló angol nyelvtudása eredményezte, hogy 1972-ben megválasztották az „Oszcilloszkópok" IEC albizottság elnökévé. Akkoriban — és valószínűleg azóta is — hazánkból ő az egyetlen, aki ilyen tisztséget töltött be az IEC-ben.

Mindezek mellett gondot fordított arra, hogy eredményeit írásban is közreadja: 7 szakkönyvet, számos egyetemi jegyzetet és kb. 50 cikket jelentetett meg.

Sokoldalú és alapos tudása kiterjedt az eszközökre éppen úgy, mint a rendszerekre. Nagy eszköztervezési feladat volt a tv-adók filterplexerei csatolt koaxiális rezonátor rendszereinek megalkotása. Ettől nagyon eltérő ismereteket és készségeket igényel a színes tv vagy a képújság rendszertechnikája. Ferenczy Pál azon kevesek közé tartozott, aki mindkét terület mesterfokú ismerője és alkotója volt.

Igazi mérnökként mindig felismerte a szakterület időszerű feladatait. E feladatok megoldását tűzte ki célul és a célokat meg is valósította. Ezekből írta publikációit. Nem azért dolgozott, hogy publikálhasson, hanem valóságos feladatokat oldott meg olyan szinten, hogy az mások számára tanulságul közreadandó lett.

Az elmúlt évtizedek során, akik Ferenczy Pállal kapcsolatban voltak, tapasztalták nagy műveltségét, sajátos humorát, figyelmességét, gondoskodását és segítségre mindig kész magatartását.

Szebbé, jobbá és értékesebbé tette az életünket.

Az élő ittléte közöttünk véget ért, de munkássága tovább él alkotásaiban és tanítványai (valamennyiönk) gondolkodásában, eredményeiben. Az utódok próbáljanak hűen helytállni, ahogyan azt Ferenczy Pál egyedülállóan, egész életével tette.

BARÁT ZOLTÁN

SAILCAD

SAIL-CAD Komputeres Tervező Rendszerek Kft.
1119 Budapest, Kevehàza u. 15-17.
Telefon: 166-9488 Telefax: 166-2867

AutoCAD munkahelyek 3 év garanciával, akciós áron!

ElonoX ECONOMY 299.000,- Ft
Elonex 386sx/25 MHz, 2 MB, 50 MB winch., mat. kooprocesszor
14"SVGA szines mon, MS-DOS 5.0, Windows 3.1, Elonex egér,
AutoCAD Rel. 11 + AME

ElonoX STANDARD 449.000,- Ft
Elonex 386/40 MHz, 4 MB, 100 MB winch., mat. kooprocesszor
17" 1280x1024 mon, DOS 5.0, Windows 3.1, Elonex egér,
AutoCAD Rel. 11 + AME

ElonoX PROFESSIONAL 699.000,- Ft
Elonex 486/33 MHz, 8 MB, 100 MB winch.
20" 1280x1024 SONY mon., DOS 5.0, Windows 3.1, Elonex egér,
AutoCAD Rel. 11 + AME

Kurta XLP 29.900,- Ft
XLP 1212, 12"x12"-os digitalizáló tábla, bármely konfigurációhoz felár

* Az árak az ÁFA-t nem tartalmazzák.

MUTEX CSOPORT A megtérülő biztonság

SYSTEM CENTER DEALER

IBM PS/2 30-H31
286
1.44 MB FDD
30 MB HDD
VGA
AT BUS
34.900 Ft+Áfa

IBM PS/2 55-X61
386 SX
1.44 MB FDD
60 MB HDD
VGA
Micro Chanel
69.900 Ft+Áfa

Olivetti Quaderno notebook
az 1 kg-os titkárnő
beépített diktafonnal és
menedzser kalkulátor
funkciókkal
DOS 5.0
109.900 Ft+Áfa

AMÍG A KÉSZLET TART

Címünk: Budapest, Lánchíd u. 7-9. Tel.: 201-6688, Fax.: 202-6367

TÁJÉKOZTATÓ SZERZŐK RÉSZÉRE

A folyóirat egyes számai az elektronika egy-egy fontos témaköréről adnak átfogó képet. A tematikus cikkeken kívül a folyóiratnak a következő állandó rovatai vannak:

- EGYEDI CIKKEK: a kitűzött témakörön kívüli cikkek számára.
- TERMÉKEK–SZOLGÁLTATÁSOK: eszközökről, berendezésekről, szoftvertermékekről és szolgáltatásokról közöl információt.
- GAZDASÁG–KUTATÁS–OKTATÁS: gazdasági összefüggésekről, kutatási lehetőségekről, szakemberképzésről ad tájékoztatást.
- HÍREK–ESEMÉNYEK: elektronikai vállalatokról, fontosabb rendezvényekről számol be.
- NÉZETEK–VÉLEMÉNYEK: az olvasók észrevételeit, megjegyzéseit közli.

A cikkeket két példányban kell beküldeni a lap felelős szerkesztőjének címére (lásd a belső borítón). A cikkek max. terjedelme 30, kettes sortávolságú gépelt oldal (minden ábrát 1 oldalnak számolva), a cikk elején 100–200 szavas magyar és angol nyelvű kivonattal. A szerzők rövid életrajzát és kontrasztos fényképét mellékelni kell. A TERMÉKEK–SZOLGÁLTATÁSOK és a GAZDASÁG–KUTATÁS–OKTATÁS rovatok cikkei legfeljebb 16, kettes sortávolságú oldal terjedelműek lehetnek.

Az ábrák tussal, fehér papírra készített eredeti példányát kell mellékelni. Az ábrákon nagybetűs feliratokat kell alkalmazni olyan méretben, hogy azok az ábrák egy vagy két hasábos kicsinyítése esetén is jól olvashatók legyenek. Az ábrafeliratokat külön lapon kell mellékelni. Lehetőség szerint kerülni kell a fényképek használatát.

A TERMÉKEK–SZOLGÁLTATÁSOK rovatban megjelent cikkekért a szerző vállalatától nyomtatott oldalanként 8000,– Ft költségtérítést kérünk.

EUROPA TELECOM'92

Budapest, 1992. október 12 – 17.

A Nemzetközi Távközlési Unió fontos és különösen időszerű eseménye kerül megrendezésre Budapesten. A Távközlési Világ Fórum előadásai a távközlés politikai, műszaki és gazdasági kérdéseivel foglalkoznak. Az előadásokkal egyidőben a távközlő berendezések gyártói 16 000 m^2 területen mutatják be termékeiket és szolgáltatásaikat.

A TELECOM'92 lehetőséget ad nagy vállalatok vezetőivel való találkozásra, a kritikus törvényi szabályozások megismerésére, közös vállalatok és beruházások előkészítésére.

EUROPA TELECOM'92
Budapest, 1992. október 12–17.

Vasárnap okt.11.	Hétfő okt.12.	Kedd okt.13.	Szerda okt.14.	Csütörtök okt.15.	Péntek okt.16.	Szombat okt.17.
Sajtó nap	Hivatalos megnyitó		KIÁLLÍTÁS Nyitva 9.30-tól 17.30-ig			
TELECOM'92 Fogadás	Kiállítás 12.00–17.30					
	POLITIKAI SZIMPÓZIUM 13.45–17.30	POLITIKAI SZIMPÓZIUM 9.00–17.30	POLITIKAI SZIMPÓZIUM 9.00–12.15			
			MŰSZAKI SZIMPÓZIUM 13.45–17.30	MŰSZAKI SZIMPÓZIUM 9.00–17.30	MŰSZAKI SZIMPÓZIUM 9.00–17.30	MŰSZAKI SZIMPÓZIUM 9.00–17.30
			GAZDASÁGI SZIMPÓZIUM 9.00–18.00	GAZDASÁGI SZIMPÓZIUM 9.00–18.00	GAZDASÁGI SZIMPÓZIUM 9.00–18.00	

FRANCE TELECOM
KÖZELEBB HOZZA A VILÁGOT

Kifejlesztettük a világ első numerikus hálózatát és a legnagyobb csomagkapcsolt adatátviteli rendszert. Nevünkhöz fűződik a legfejlettebb és legismertebb videotex és az első országos RNIS (integrált szolgáltatású digitális hálózat). Ezek a France Telecom sikerei. De a technológia – bármennyire is csúcsteljesítmény – csak akkor válik értékké, ha testre szabott.

Ezért az Ön telekommunikációs rendszerét is közösen fogjuk megteremteni és megvalósítani. Hogy közelebb hozzuk a világot.

További információkért hívja:

Párizsban	*(33 1) 44 44 57 06*
Budapesten	*(36 1) 201 2579*
Moszkvában	*(7 095) 238 66 48*
Varsóban	*(48 22) 17 65 26*
Prágában	*(42 2) 220 786*

EUROPA TELECOM 92

B PAVILON
FRANCIA STAND

Párizs • London • Bonn • Frankfurt • Madrid • Brüsszel • Moszkva
Varsó • Prága • Budapest • New York • Chicago • San Francisco
Mexikóváros • Buenos Aires • Tokió • Peking • Dzsakarta • Szingapúr

FRANCE TELECOM

★ RSCG HAVAS & VARGA