

Szakdolgozat

Zsiros Szabolcs

Debrecen

2008

Debreceni Egyetem
Informatikai Kar

**ADAT- ÉS HANGÁTVITEL
KÁBELTELEVÍZIÓS HÁLÓZATOKON**

Témavezető:
dr. Almási Béla
Egyetemi docens

Készítette:
Zsiros Szabolcs
Programtervező informatikus

Debrecen

2008

TARTALOMJEGYZÉK

1. Bevezetés.....	4
2. A rendszer felépítése, komponensei	5
3. A technológiáról általánosan	9
3.1. A DOCSIS szabvány kialakulásának körülményei	9
3.2. DOCSIS verziók	10
4. A rendszer követelményei	11
4.1. Fizikai réteg	11
4.1.1. Modulációs technikák.....	11
4.1.2. A kábelhálózattal szemben támasztott követelmények	14
4.2. Adatkapcsolati réteg	18
4.3. Hálózati réteg.....	21
4.4. Applikációs réteg	23
4.4.1. A ToD szolgáltatás	23
4.4.2. A TFTP protokoll	24
4.4.3. A hálózat monitorozása	26
4.4.3.1. Az SNMP protokoll.....	27
4.4.3.2. A Nagios rendszer	30
4.4.3.3. MRTG.....	32
5. A hangszolgáltatás eszközei	34
5.1. A PacketCable architektúra	34
5.2. SIP	37
5.3. Az Asterisk PBX	40
6. Összefoglalás	43
7. Szójegyzék	44
8. Irodalomjegyzék	50

1. Bevezetés

Az internetszolgáltatás a szélessávú adatátvitelt lehetővé tevő technológiák megjelenésének köszönhetően igen széles körben elterjedté vált a világban. A szélessávú internet eléréssel rendelkező háztartások jelentős hányadába a világon igen elterjedt és népszerű DOCSIS technológia segítségével, kábeltelevíziós hálózatokon keresztül jut el a világháló.

Az utóbbi másfél évben volt szerencsém több, kisebb-nagyobb internet szolgáltatóval közvetve együttműködni, illetve dolgozni. Ennek köszönhetően lehetőségem nyílt a kábeltelevíziós hálózat, valamint az ezen keresztül megvalósított adat- és hangátviteli szolgáltatások, technológiák közelebbi megismerésére.

Éppen ezért kézenfekvő volt, hogy erről a témáról írjam a szakdolgozatomat.

A szakdolgozatom célja a DOCSIS hálózatokon alapuló szolgáltatások üzemeltetéséhez szükséges alapvető követelmények, technológiák, eszközök, protokollok összegyűjtése és ismertetése. A terület igen sokrétű és összetett mivolta miatt, az összes érintett protokoll, eszköz, stb. ismertetése messze túlmutat ezen dokumentum keretein, ezért ezek a teljesség igénye nélkül kerülnek bemutatásra. Továbbá az írásom az alapvető számítógépes hálózati ismeretek meglétét feltételezi az olvasótól.

A szakdolgozatom első fejezete hivatott betekintést engedni a kábeltelevíziós internetszolgáltatás hálózatának felépítésébe, illetve ismertetni az azt alkotó eszközök funkcióját.

A második nagy, „A rendszer követelményei” című fejezetét a hibrid hálózati referenciamodell mentén építettem fel, haladva a hálózat fizikai összetevőitől, követelményeitől a különféle, a rendszerhez szervesen kapcsolódó, illetve azt kiegészítő alkalmazások felé. Fontos megjegyezni, hogy ebben a fejezetben elsősorban a DOCSIS 1.x és 2.0 verziója szolgál alapul. Bár már létezik a 3.0-ás verzió, az alapjaiban nem tér el elődeitől, inkább csak funkcionalitásában bővült.

Az utolsó nagyobb lélegzetvételű fejezet a hangszolgáltatás alapjait, alapvető összetevőit hivatott ismertetni, bár ez a témakör önmagában egy külön szakdolgozatot érdemelne.

A dőlt betűvel jelölt kifejezések a mellékelt Szójegyzékben kerülnek elmagyarázásra.

2. A rendszer felépítése, komponensei

A kábeltelevízió az egyik legelterjedtebb médium, amely megjelenik családi házakban, lakásokban, cégeknek, és általánosságban a modern világban.

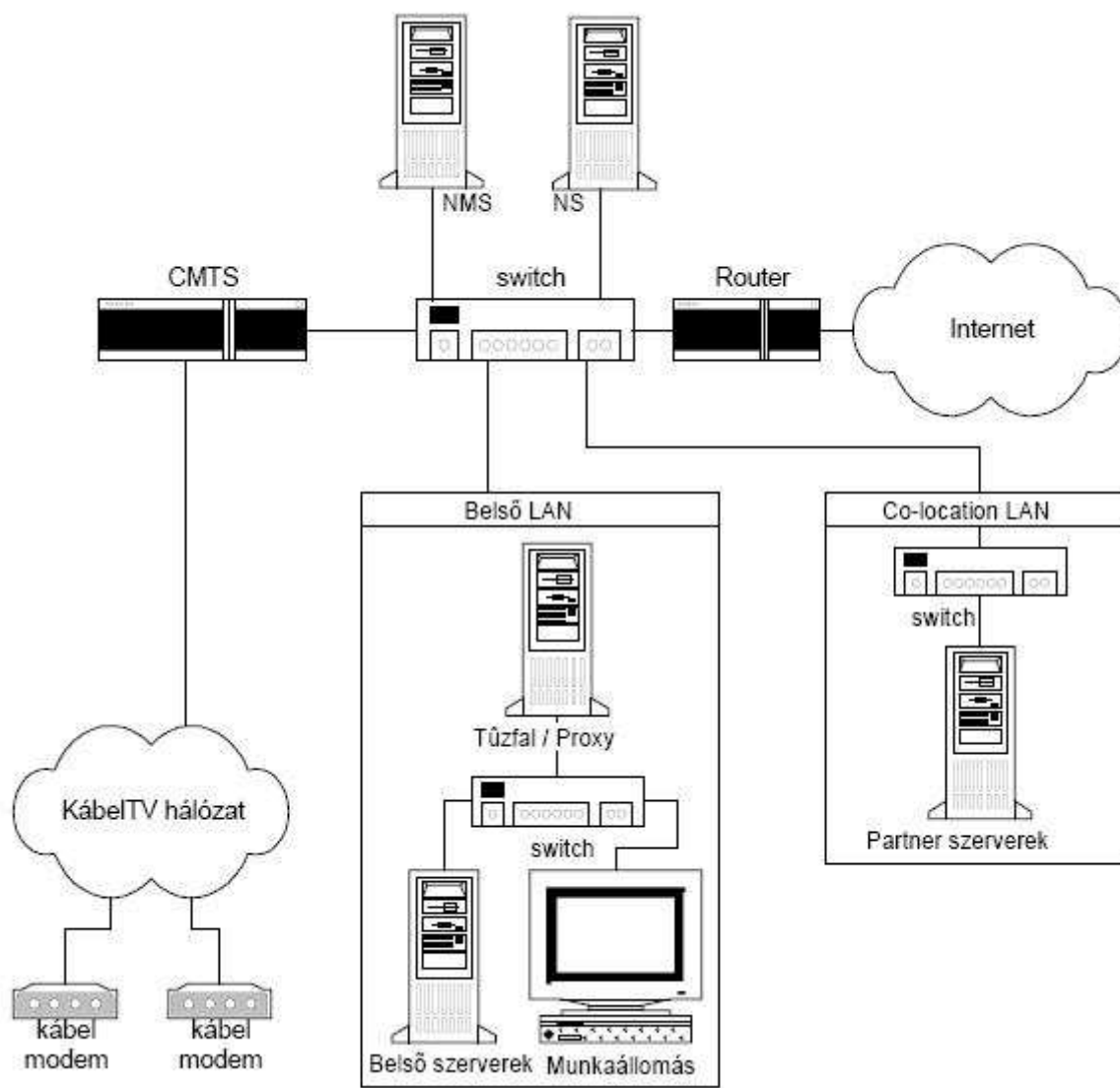
A kábeltelevíziós szolgáltatás az 1900-as évek közepén azon alapötletből indult ki, mi szerint nincs szükség minden lakásban külön antennákra a televízióadások vételéhez. Elég lenne közösségenként egyetlen jó minőségű antenna és az általa fogott jelet, erősítést követően veszteség nélkül el lehetne juttatni a végfelhasználókhoz. Ezt hívják angol rövidítéssel *CATV*-nek. Ez jelentősebb anyagi ráfordítás nélkül kivitelezhető volt, így kifejezetten elterjedté vált azon helyeken, ahol a földrajzi viszonyok miatt nem lehetett jó minőségben fogni az adást.

Ezt követően később ezeket az elszeparált televíziós hálózatokat egyetlen nagy rendszerre kapcsolták össze.

Azonban továbbra is szükség volt területenként egy fejállomásra, ahová beérkezik a televíziós csatornák jele, illetve az internet gerincvonala és ahonnan ezek továbbítódnak az előfizetők felé. Ezeken a fejállomásokon található a szolgáltatáshoz nélkülözhetetlen központi eszközök, mint például a *CMTS*(ek), a különféle kiszolgáló szerverek, stb.

Az internet gerincvonala, melyből akár redundáns módon több is beérkezhet a kábelszolgáltató fejállomására, egy routeren keresztül kapcsolódik a szolgáltató belső hálózatára. Ezen router feladata a forgalom irányítása a belső hálózaton belül, és azon kívül, az internet felé. A fő internetvonal megszakadása esetén az ő feladata a tartalékvonal felé irányítani a forgalmat.

Az alábbi ábrán *NS*-ként jelölt névszerver feladata kettős. Egyrészt hiteles választ adó névszerverként kezelnie kell a helyi domaint. Másrészt a külvilág felé tartó forgalom meggyorsítása érdekében cache funkciókat is el kell látnia, azaz a korábbi névfeloldási kérelmekre érkezett válaszokat bizonyos ideig tárolnia kell, és az így tárolt eredményeket kell szolgáltatnia a későbbi ugyanarra a névtérre irányuló kérések esetén.



1. ábra. A kábelszolgáltató teljes funkcionalitású belső hálózata

A *CMTS* teszi lehetővé a kábelmodemek egymással, és a külvilággal való kommunikációját. Feladata a kábelmodemek a hálózatra való feljelentkezésének levezénylése, valamint az adatátvitel lebonyolítása. Ez egy igen összetett feladat, hiszen a koaxiális *RF* interfészeire csatlakoztatott kábelhálózat, mint adatátviteli közeg egy elosztott média, amelyen egyszerre több kábelmodem is megjelenhet adó szerepkörrel. Ezért a *CMTS* egyik legfontosabb tevékenysége magának a közeghozzáférésnek a vezérlése. Ő látja el, kontrollálja a hibrid referenciamodell alsó két rétegbe tartozó feladatok nagy részét, szerepköre elsődlegesen erre a két rétegre terjed ki.

A kábelmodem egy olyan eszköz, mely kapcsolatot teremt a végfelhasználó internetre csatlakoztatni kívánt eszköze és a kábelhálózat között. Egymagában, a *CMTS* vezénylete nélkül semmilyen használható funkciót nem tud nyújtani, és más kábelmodemekkel sem tud kommunikálni. Bekapcsolása után ezért rögtön a *CMTS* jeleit kezdi el keresni a kábelhálózaton, majd miután megtalálta, a kapott utasításoknak megfelelően felkészíti magát a felsőbb szintű, kétoldali adatkommunikációra.

Az *NMS* szerver szerepe az architektúrában igen komplex. Egyrészt nyújtania kell a különféle felsőbb rétegbeli szolgáltatásokat, melyeket a *DOCSIS* eszközök igényelnek. Ilyenek például a *ToD*, a *TFTP*, vagy a *DHCP* szolgáltatások. Ezen kívül tevékenységi körébe tartozik még a különféle hálózatmenedzselési feladatokhoz támogatást nyújtani, monitorozni az eszközöket, valamint naplózni a hálózatban történő eseményeket.

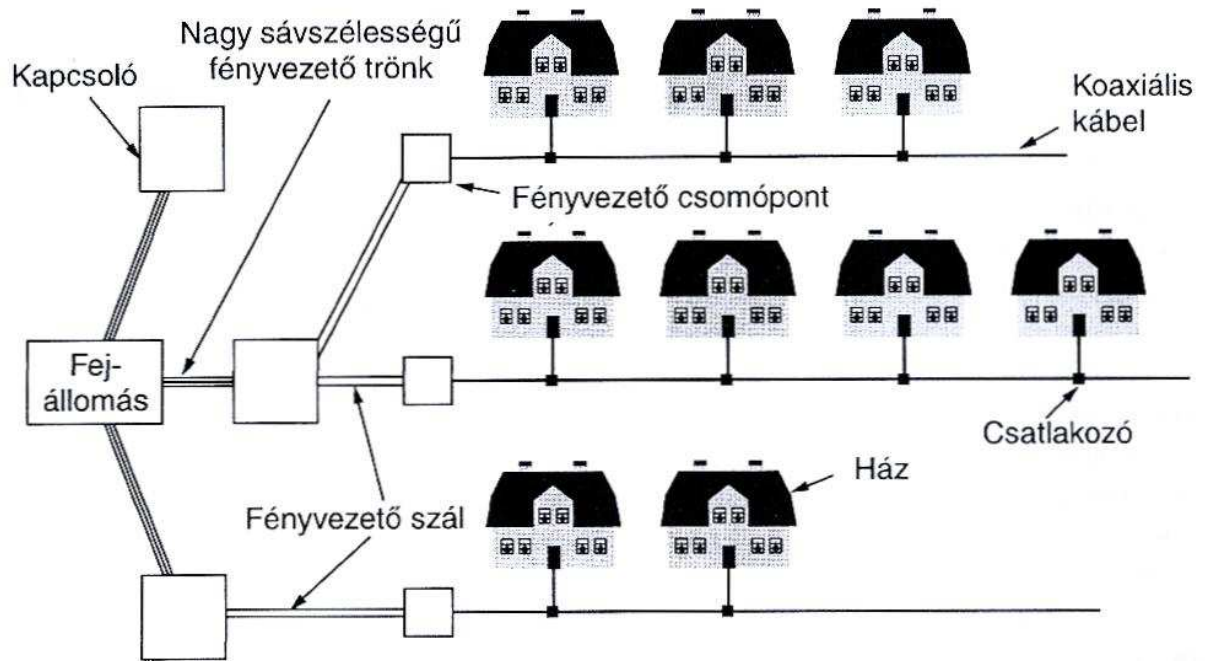
A most említett eszközök, szerverek csak az alapvető és legszükségesebb funkciókat nyújtják. További rendszerekre lehet szükség az egyéb szolgáltatások nyújtásához, amelyek ma már minden internet szolgáltatóval szemben alapkövetelménynek minősülnek, mint például a saját email cím és webtárhely adása az ügyfeleknek.

Ezen kívül belső alhálózat kialakítására is szükség lehet a fejállomáson a helyi munkaállomások kiszolgálására. Valamint további, feláras szolgáltatásként az internet szolgáltatók az ügyfelek, partnereik szervereit is elhelyezik a fejállomáson. Az 1. ábrán ez utóbbi két szolgáltatási területet hivatott jelölni a bekeretezett rész.

A kábelhálózat maga úgynevezett *HFC* hálózat, ami azt jelenti, hogy a hálózatban fényvezető és koaxiális kábelek egyaránt megtalálhatók. Ennek gyakorlati oka, hogy a fényvezető kábelnek jóval nagyobb a sáv szélessége a koaxiálishoz képest, így az több koaxiális alhálózatot is táplálhat. Az optikai kábelek úgynevezett *optikai node*-okba, optikai csomópontokba csatlakoznak, aminek a feladata gyakorlatilag az adatátviteli közegváltás. Ezen *node*-okból továbbítódik a jel erősítőkön át az előfizetői végpontokba. A *HFC* hálózat további előnye a tisztán koaxiális megoldással szemben az optikai vezetők zavartűrőbb volta.

Volt szerencsém élesben találkozni mindkét rendszerrel és az előbbi látványosan bebizonyosodott. A tisztán koaxiális hálózat elfogadhatónak mondható 22-25ös *USSNR*-je

(a hálózat fizikai méréséről később esik szó) az átépítés után a kiváló 35-ösre emelkedett. Általánosságban elmondható, hogy minél nagyobb a hálózat optikai részének a koaxiálissal szembeni aránya, annál zaj-mentesebb a hálózat.



2. ábra. A HFC hálózat

3. A technológiáról általánosan

3.1. A DOCSIS szabvány kialakulásának körülményei

A kábelhálózaton történő internetszolgáltatás korai éveiben nem volt egységesített szabvány a hálózat működését, illetve a kábelmodemeket illetően. Ez a szolgáltatók számára előnyt jelentett, hiszen a saját modemüket bérbe adva az előfizetőknek jelentős mennyiségű bevételt könyvelhettek el. Ennek ellenére hamar egyértelművé vált, hogy célszerű lenne egy nyílt szabvány kidolgozása, mely piaci versenyhelyzetet teremtené a kábelmodemek gyártói közt és az így bekövetkező árcsökkenés elősegítené a szolgáltatás terjedését.

1994. májusában több nagyvállalat részvételével létrejött az IEEE 802.14 Cable TV Media Access Control (MAC) and Physical (PHY) Protocol Working Group elnevezésű albizottsága, melynek feladata volt, hogy létrehozzon egy egységes nemzetközi adatátviteli szabványt a HFC, azaz a hibrid fényvezető-koaxiális kábeltelevíziós hálózatokon. A bizottság egy olyan szabványt kívánt alkotni, mely a legkorszerűbb technológiákat alkalmazva időtálló. A terv igen nagyszabású volt, miatt az eredetileg 1995. decemberéig megszabott kidolgozási határidő kitolódott 1997 végére.

Mivel azonban a kábelrendszer üzemeltetők elsődleges szempontja az újonnan létrehozandó szabvány mielőbbi üzembe állítása volt, nem vártak tovább az IEEE 802.14-re. 1996. májusában létrehozták a Multimedia Cable Network System nevű társaságot a Comcast, a Cox, a TCI és a Time Warner részvételével. Később hozzájuk csatlakozott a MediaOne, a Rogers CableSystem és a CableLabs cég is. Az így létrejött szövetség magába foglalja az Észak-Amerikai kábeltelevíziós szolgáltatók nagytöbbségét.

A lehető legkisebb költségekre és a mielőbbi piacra dobásra törekedve a szövetség 1997. márciusában kiadta a *Data Over Cable Service Interface Specification*-t, röviden a DOCSIS-t. A szabvány sikeresnek bizonyult, így nem sokkal később elkészítették annak európai verzióját is, az *EuroDOCSIS*-t, melyet az európai kábeltelevíziós szabványokhoz igazítottak. A két terület szabványának különbségeiről később folyamatosan említést teszek.

3.2. DOCSIS verziók

A *DOCSIS* szabványok csak az alsó két (fizikai és adatkapcsolati) réteg leírását tartalmazzák. A *DOCSIS* első, 1.0-s szabványa az úgy nevezett „best-effort” módszert alkalmazta sávszélesség kiosztásra. Ez a gyakorlatban annyit tesz, hogy mindenki akkora sávszélességet kaphat, amennyi az aktuális terhelésnek megfelelően jut. Így ezen verzió mellett nem lehetett különféle díjsomagokat sem definiálni, hiszen nem lehetett korlátozni a sávszélességet sem. Ezért, és az olyan időkritikus szolgáltatások miatt, mint a *VoIP* kellett továbbfejleszteni a szabványt. Így jött létre 1999-ben a *(Euro)DOCSIS 1.1*, amely már támogatta a szolgáltatás minőségét meghatározó *QoS*-t is. A szabvány ezen továbbfejlesztése nem jelentett a szabványt már régebben használó szolgáltatók számára többletköltséget, hiszen az átálláshoz csak egy *firmware*-frissítésre volt szükség.

A 2001-ben debütáló *DOCSIS 2.0* jelentős fejlődést hozott visszirányú sebesség és stabilitás tekintetében, lehetővé téve a szimmetrikus szolgáltatást is. Ehhez nyilvánvalóan módosítani kellett a fizikai réteget is, ami azonban nehezebbé és költségesebbé tette a verzióváltást.

2006. augusztusában jelentették meg a 3.0-s verziót, mely nagy mértékű javulást hozott mind az előre-, mind a visszirányú sebesség tekintetében, valamint olyan újításokat tartalmaz, mint például a *channel-bonding*, vagy az *IPv6* és az *IPTV* támogatása.

4. A rendszer követelményei

Ezen fejezet célja, hogy ismertesse a kábeltelevíziós hálózaton keresztüli internetszolgáltatáshoz szükséges alapvető technikai és szoftveres feltételeket. Ennek talán legcélszerűbb megközelítése, ha az ötrétegű hibrid referenciamodell mentén haladva vesszük végig az említetteket, hiszen egy kialakítandó szolgáltatói rendszer tervezése is a fizikai hálózat kialakításával kezdődik, majd csak ennek ismeretével állíthatók be az egyre szoftver közelebbi részek.

Az applikációs réteghez eljutva rengeteg szolgáltatásról, szoftverről, illetve protokollról lehetne szót ejteni, azonban csak az általam legszükségesebbnek vélt, a szolgáltatás biztosításához szorosan kötődő elemek kerülnek górcső alá.

4.1. Fizikai réteg

4.1.1. Modulációs technikák

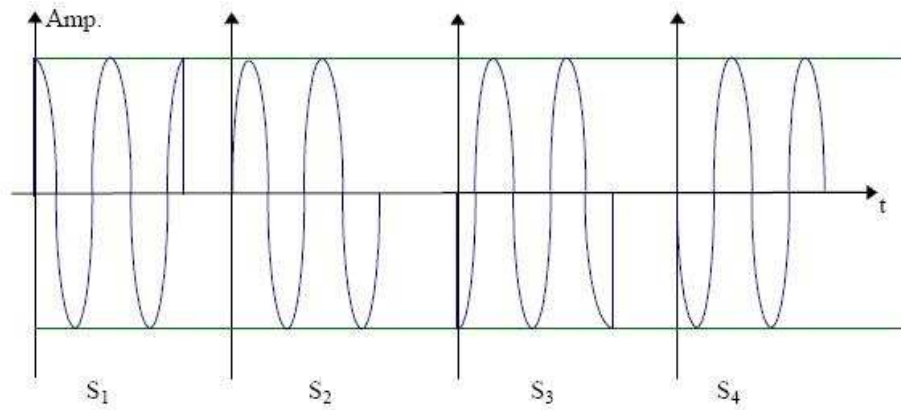
Azt a folyamatot, amely során a digitális információt, analóg vivőjelre képezzük le, modulációnak nevezzük.

A moduláció során az analóg jellel ellentétben nem az alakhűség az elsődleges szempont, hanem a minél kisebb hibavalószínűség az átvitel során. A televíziós csatornák továbbítására frekvenciaosztásos multiplexelésű közeghozzáférést alkalmaznak (röviden *FDMA*), ezért vivős modulációs technikát alkalmazunk a továbbításra.

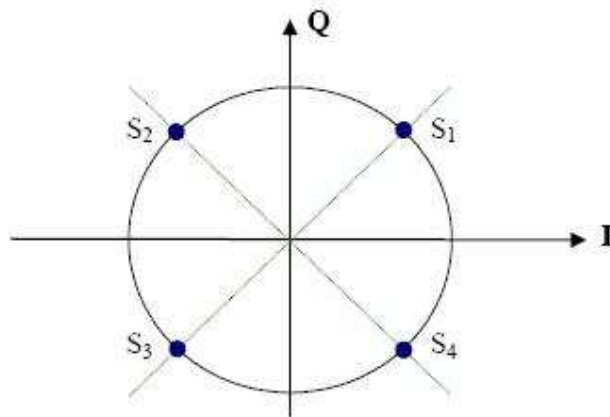
Az analóg csatornához hasonlóan az adatátviteli csatornákat is középfrekvenciájukkal szokás jellemezni. A jel fázisa és/vagy amplitúdója hordozza a digitális információt. Egy állapotot (szimbólumot) egy adott amplitúdó és fázishelyzet jelöl. Hogy nagy adatátviteli sebességet érjünk el, többállapotú modulációt kell alkalmaznunk. Az 1 másodperc alatt végbemenő állapotváltozások számát szimbólumsebességnek nevezzük. Mértékegysége a MSymb/s, azaz a megszimbólum másodpercenként.

A digitális modulációs technikák részletes leírása az [5] dokumentumban olvasható.

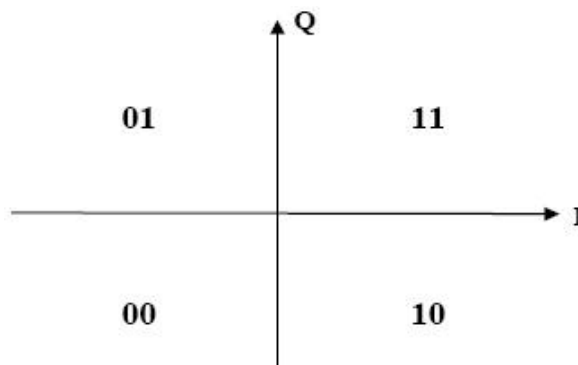
QPSK moduláció: Ez egy úgynevezett négyállapotú fázismoduláció, mely során a jel amplitúdója nem, csak a fázisa változik a négy állapot között. Ezt a modulációs technikát robusztussága miatt elsősorban vissz irányban szokás alkalmazni.



3. ábra. A QPSK jelkészlete

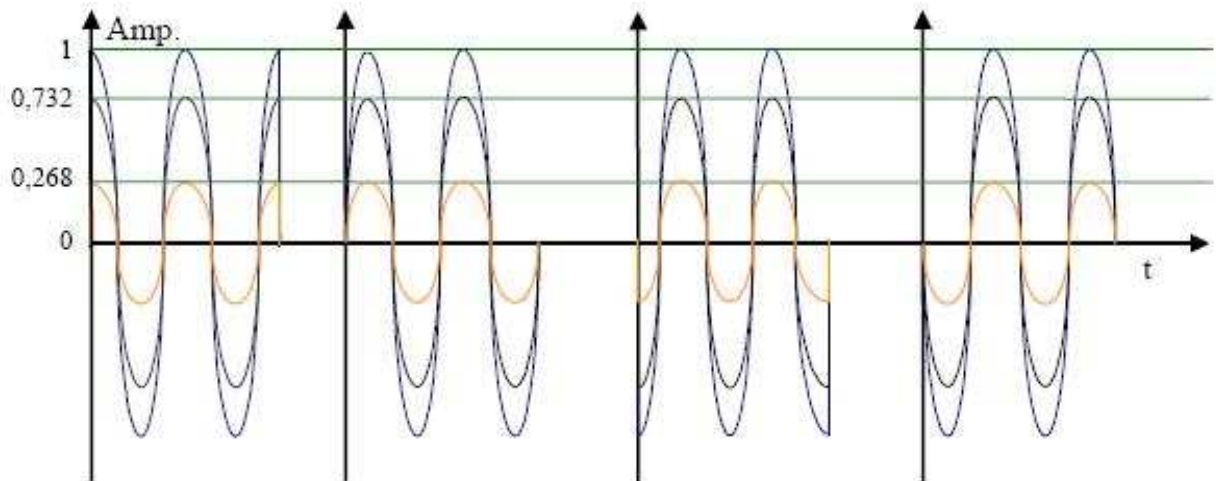


4. ábra. A QPSK vektorképe

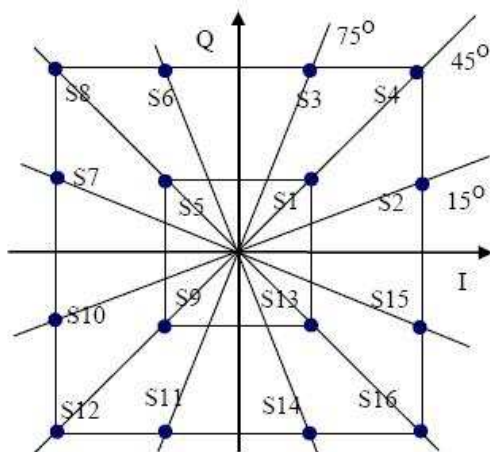


5. ábra. A QPSK bittérképe

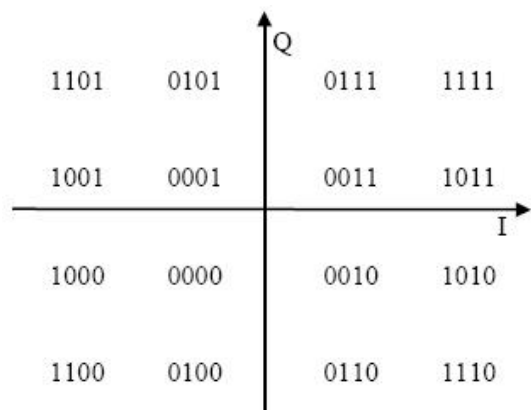
QAM moduláció: Ezen moduláció esetében a vivőfrekvencia amplitúdója és fázisa is változik. E két összetevő megengedett értékeitől függően beszélünk *16QAM* (4-4 szint), *64QAM* (8-8 szint), és *256QAM* (16-16 szint) modulációról. Léteznek további *QAM* modulációk is, de kábeltelevíziós szempontból csak ezek érdekesek.



6. ábra. A 16QAM jelkészlete



7. ábra. A 16QAM vektorképe



8. ábra. A 16QAM bittérképe

A *16QAM* igen zavartűrő, így elsősorban visszirányban, míg a *64QAM* és a *256QAM* modulációkat nagy adatátviteli sebességük miatt előreirányban használják.

Egy bit átviteléhez két állapot kell, a 0 és az 1. *QPSK* modulációval tehát 2 bit vihető át szimbólumonként, *16QAM*-mel 4 bit, *64QAM*-mel 6 bit, míg *256QAM*-mel 8 bit.

4.1.2. A kábelhálózattal szemben támasztott követelmények

A kétirányú adatátvitel megvalósításához meg kell adnunk, be kell állítanunk a *downstream* és az *upstream* irányú kommunikációs csatornák frekvenciaszintjeit.

Mivel letöltési irányban általában nagyobb, feltöltési irányban kisebb forgalom zajlik, ezért az általuk használt frekvenciatartományok is hasonlóan vannak elosztva. A *DOCSIS* szabvány 88~860 MHz, míg az *EuroDOCSIS* 108~862 MHz között határozza meg a *downstream*, valamint 5~42 MHz, illetve 5~65 MHz között az *upstream* csatornák frekvenciatartományát. Itt jelentkezik az egyik jelentős különbség a két szabvány között, melynek oka az eltérő televíziós szabványokban keresendő.

A megadott értékeken belül természetesen nem helyezhetők el akárhol a *downstream* és az *upstream* csatornák, hiszen a *downstream* tartomány és a televíziós műsorszórásra használt frekvenciatartomány átfedésben van egymással, így ügyelni kell a csatornakiosztásra, valamint figyelembe kell venni azt is, hogy az FM rádiós közvetítések is a 88~108 MHz-es tartományban helyezkednek el.

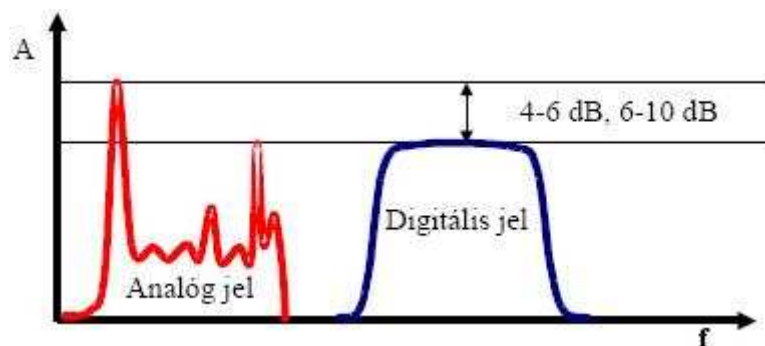
A két irány működésében, és behangolásában alapvető eltérések vannak, ezért nézzük most külön-külön őket.

Előreirány: A *DOCSIS* és az *EuroDOCSIS* másik alapvetően különbsége szintén a kábeltelevíziós szabványok miatt adódik. A főleg Észak-Amerikában elterjedt *NTSC* 6, míg az Európában domináló *PAL* és *SECAM* szabványok az előbbihez képest jobb képminősége miatt 8 MHz-es csatorna sávszélességet használnak. Természetesen az *(Euro)DOCSIS* szabványoknak is alkalmazkodniuk kellett ehhez, és ebből következik az *EuroDOCSIS* előnye a párjához képest. A nagyobb csatorna-sávszélesség ugyanis nagyobb adatátviteli sebességet eredményez.

Az előreiránnyal az üzemeltetés során alapvetően kevés probléma adódik. A jel folyamatosan, gyakorlatilag állandó szinttel jelen van, így az esetleges hibákat is könnyű feltárni. Egy egyszerű jelteljesítmény mérés a legtöbb esetben elég, de egy *MER* méréssel a moduláció minőségét is leteztelhetjük.

Az előreirányú frekvencia a megadott határértékeken belül tetszés szerint elhelyezhető (természetesen a korábban említetteket figyelembe véve), még nem szabványos csatornán is. A modem bárhol megtalálja azt, bár nem szabványos csatornán jelentősen lassabban.

A 64QAM modulációjú digitális előreirányú jelek jelszintjét 6-10 dB-lel, míg a 256QAM modulációjú jelekét 4-6 dB-lel az analóg vivők szintje alá kell helyezni, mivel a túlságosan magas jelteljesítmény torzításokat okoz a többi csatornán. A digitális jelek spektrumterhelése és a digitális vevők toleranciája a vétel szintjét tekintve igen jó, így ez a különbség nem jelent gondot a modemek számára.



9. ábra. Előreirányú jelszint

	Vétel jelszint	Szabvány SNR	Mérhető SNR		MER	
			64 QAM	256 QAM	64 QAM	256 QAM
DOCSIS	-15 ~ +15 dBmV	35 dB	23,5 dB	-	28 dB	32 dB
	-15 ~ -6 dBmV		-	33		
	-6 ~ 15 dBmV		-	30		
EuroDOCSIS	-17 ~ +13 dBmV	44 dB	25,5 dB	-	28 dB	32 dB
	-13 ~ -6 dBmV		-	34,5 dB		
	-6 ~ +17 dBmV		-	31,5 dB		

10. ábra. Az előreirány követelményei

A fenti táblázat foglalja össze a kábelmodem csatlakozási ponttal szemben támasztott követelményeket. Ha ez nem teljesül egy végpontra, a rajta levő kábelmodem nem fogja tudni

megtalálni a *downstream* csatornát. Ha azonban ennek megfelelőek az értékeket, akkor az előreirány biztosan jól működik.

A vételi jelszint megadott határok közé esését könnyen ellenőrizhetjük egy jelteljesítmény mérővel.

Egy igen fontos és informatív jellemző a *MER* érték, azaz a modulációs hibaarány. Ezen jellemzőnek a mérése gyakran sokkal jobban használható, mint a jel-zaj viszonyé, hiszen ennek mért értéke nem függ a használt szabványtól, kizárólag csupán a modulációtól.

Összességében elmondható, hogy ha jó minőségűek a hálózatban használt eszközök, és az előreirány jól be van állítva, akkor azzal nem sok gond jelentkezhet hosszú távon sem.

Ha a rendszert a későbbiekben bővíteni szükséges, akkor az vagy egy újabb *downstream* frekvencia kijelölésével, vagy az előreirányú ágak fizikai szétválasztásával tehető meg. Ez utóbbi esetben ügyelni kell, hogy a *downstream*-ekhez tartozó *upstream* csatornák is a megfelelő hálózatrészben legyenek.

Visszirány: Az (*Euro*)*DOCSIS* szabvány a szükséges *SNR*-en kívül nem ad további útmutatást a visszirány beállításához, pedig a visszirány, összegző jellegű hálózatról lévén szó, a legkritikusabb pontja a rendszernek. A sok kimenet felől érkező jel összeadódva jelenik meg az egyetlen bemeneten, így ha egy előfizetői végponton zajbetörés keletkezik, akkor az végiggyűrűzhet az alhálózat további részén, megbénítva így akár egy egész körzet kommunikációját.

A kábelhálózatba bekerülő zajok túlnyomó többsége az előfizetőktől ered.

Természetesen erre is van megoldás, mégpedig a következetes szűrőzés. A nem internetező előfizetőktől már a falaljzatnál érdemes lezárni a visszacsatornát. A kábelmodemes előfizetőknél pedig célszerű *multimédiás aljzatot* felszerelni, amely több funkciót is ellát. Egyrészt splitterként funkcionálva, külön csatlakozást nyújt a TV, a rádió és a kábelmodem számára, másrészt csak a modem felől engedi át a visszirányú frekvenciasávot. Fontos információ a multimédiás aljzattal kapcsolatban, hogy a kábelmodemes kimenetén az előreirányú jelet 10 dB-lel, a visszirányút pedig 1-1,5 dB-lel csillapítja.

A visszirány zavarmentessé tétele mellett még be is kell színtezni, illetve állítani azt. A kábelmodem visszirányú frekvenciája és csatorna sávszélessége egy bizonyos tartományon belül szabadon állítható, viszont az adási teljesítmény automatikusan állítódik be a *CMTS* vezényletében. Ezt úgy kontrollálja a *CMTS*, hogy a bemenetén minden egyes modemtől ugyanolyan szinttel érkezzen meg a jel.

A visszirány színtezésénél az előbbi mellett két szempontot kell figyelembe venni: a modem szabályozási tartományát, illetve a hálózat aktív elemeinek optimális működését. Összességében a visszirányt úgy kell színtezni, hogy minden visszirányú erősítő és *optikai node* bemenetén egységesen 18 dBmV legyen a jelszint.

A *CMTS*-ben -16~26 dBmV-os keretek között állítható a vételi szint. Az ugyanazon csatornában lévő modemek adási jelszintjei ettől csupán 2-3 dB-lel térhetnek el. Ha ez egy adott modemre nem teljesül, akkor a *CMTS* visszautasítja a modem csatlakozását.

A *DOCSIS* és *EuroDOCSIS* szabványok 25, illetve 22 dB-ben határozzák meg az optimális *USSNR*-t. Minél magasabb a visszirányú csatorna frekvenciaszintje, annál kevésbé lesz az zajos. Ebből következik az *EuroDOCSIS* szabvány egyik előnye a *DOCSIS*-al szemben, hiszen az előbbinek 65, míg az utóbbinak 42 MHz-ig terjed a visszirányú frekvenciatartománya.

A sávszélesség 200 kHz-től 3,2 MHz-ig változtatható, és *QPSK* vagy *16QAM* modulációt lehet alkalmazni mindkét szabvány esetében. A zavartűrőbb visszirányhoz csökkenteni kell a sávszélességet és/vagy *QPSK* modulációt kell használni, de ez természetesen az adatátviteli sebesség csökkenését jelenti.

HFC hálózatoknál az optikai csomópontokból érkező jeleket (elektromos jellekké történő átalakítás után) egy splitter egyesíti, és a jel így érkezik meg a *CMTS* valamely upstream ágára. Normál eszközökkel egy upstream ágon az elérhető legjobb jel-zaj viszony 4 optikai csomópont egyesítése esetén körülbelül 25 dB lesz, amely még éppen megfelel a szabványban meghatározottaknak. Ennél több *ONU* jelének összegzése esetén a paraméterek már nem felelnének meg az előírásnak, és nem működne jól a visszirányú hálózat.

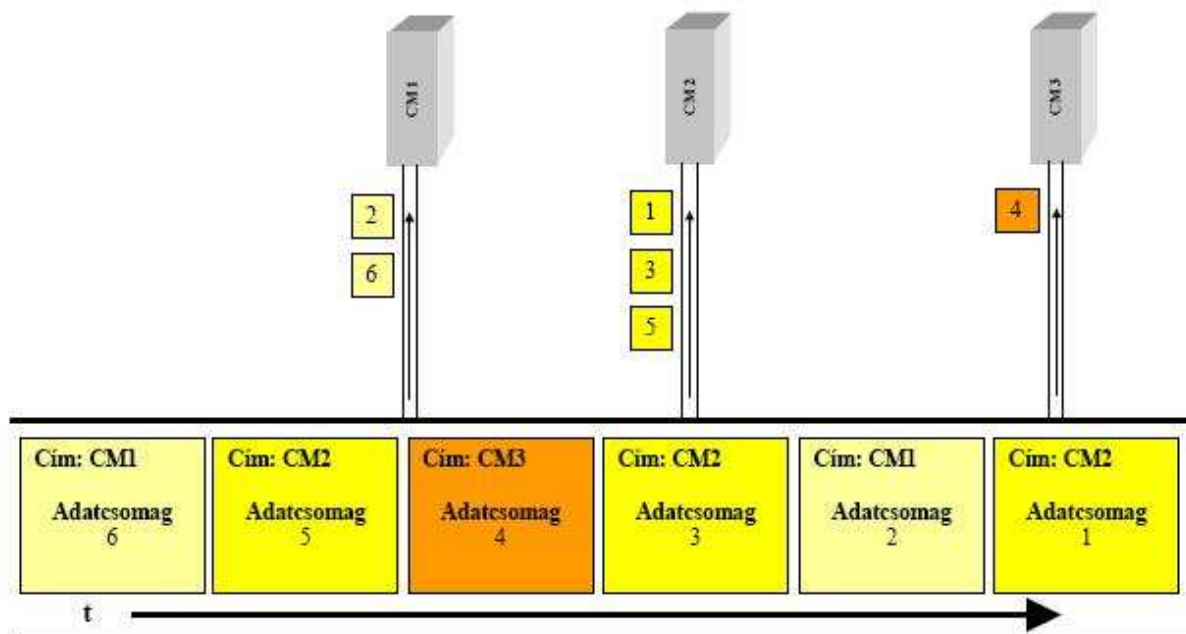
A fizikai réteghez kapcsolódóan végezetül álljon itt az egyes *DOCSIS* verziók esetén, ideális esetben elérhető, elméleti maximális adatátviteli sebességet tartalmazó táblázat.

Verzió	DOCSIS		EuroDOCSIS	
	Downstream	Upstream	Downstream	Upstream
1.x	42.88 Mbit/s	10.24 Mbit/s	55.62 Mbit/s	10.24 Mbit/s
2.0	42.88 Mbit/s	30.72 Mbit/s	55.62 Mbit/s	30.72 Mbit/s
3.0	343.04 Mbit/s	122.88 Mbit/s	444.96 Mbit/s	122.88 Mbit/s

11. ábra. Maximális átviteli sebesség a különböző *DOCSIS* verzióknál

4.2. Adatkapcsolati réteg

Ahogy az az előző fejezetből már sejthető, a kábelhálózaton az adatkommunikáció broadcast módon folyik, és a csatorna mindenkinek a csomagjait tartalmazza.



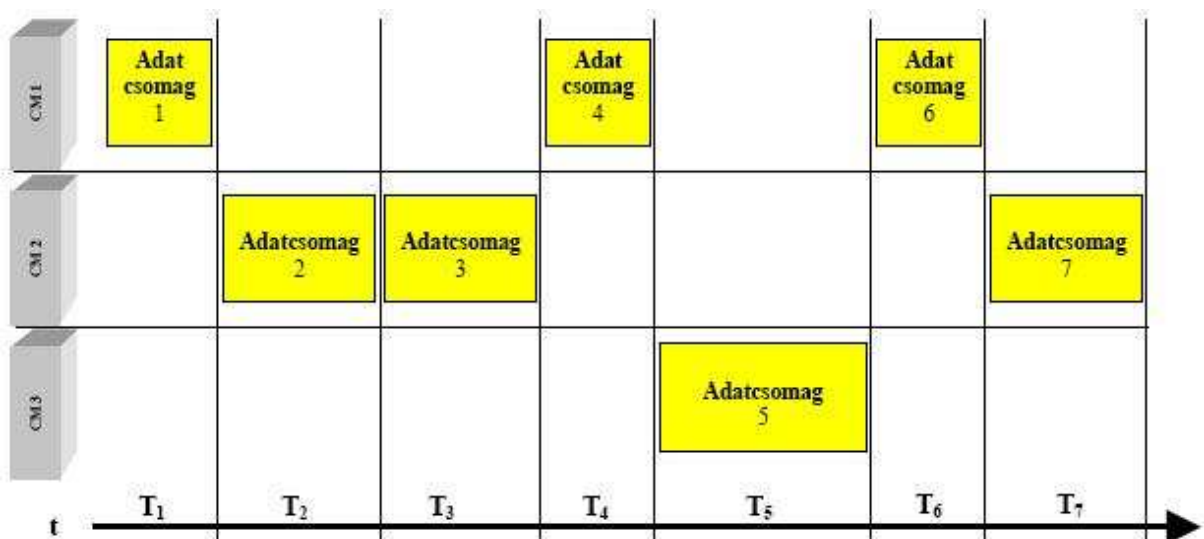
12. ábra. Előreirányú adattovábbítás

Mivel előre irányban csak egy adó van (a fejállomás), ezért itt nem merül fel probléma a közeghozzáférést illetően. A visszirány már problémásabb, hiszen mint ahogy arról már korábban szó volt, több adó (a kábelmodemek) van jelen egyszerre a hálózaton.

A konkurenciakezelést a korábbi, 1.0-s és 1.1-es *DOCSIS* szabványok *TDMA* alkalmazásával oldották meg. Pontosabban, *downstream* irányban *TDM* volt alkalmazva, az *upstream* csatornákon pedig *TDMA*. A *DOCSIS* 2.0-s verziójának tervezésekor két közeghozzáférési indítványt terjesztettek be. Az egyik a korábbi verziókból megtartott *TDMA*, a másik az *S-CDMA*. A döntés végül az lett, hogy mindkét módszert beleépítik a szabványba.

Nézzük meg most ezeket külön-külön, a [6]-os dokumentum alapján.

TDMA: Többszörös, időosztásos hozzáférés. Egyszerű és elterjedt közeghozzáférés, melyet több szabvány is alkalmaz. A különböző felhasználókhöz külön időréseket (*time slot*) rendel. Amíg egy időrés ki van osztva egy felhasználónak, addig az összes többi modem csendben marad, hogy ne legyen összeütközés. A kábel *upstream* csatornáján a *TDMA* valójában az *FDMA* és a *TDMA* kombinációja, hiszen mikor a *CMTS* időréseket oszt ki a visszirányú ágai egyikén, minden egyes *time slot* a *TDMA* módon belül több modemhez is tartozik.



13. ábra. *TDMA* a visszirányú csatornán

S-CDMA: A DOCSIS 2.0-tól kezdve alkalmazható, Terayon által kifejlesztett közeghozzáférési módszer tulajdonképpen nem egy valódi CDMA, hanem sokkal inkább a CDM, a CDMA, és a TDMA ötvözete. A bejövő adatok *minislot*-okba vannak szervezve, melynek két kiterjedése van (a továbbító kódok és az idő). A *minislot*-ok időtartama egy S-CDMA keret, amely több S-CDMA szimbólum intervallumon ível át. A maximális keret hossz 32 S-CDMA szimbólum intervallum. A szimbólumtovábbítás egy 128 részből álló sorozat általi szorzással érhető el. Ezen részek 128 ortogonális kódból álló halmazból származnak. Egy S-CDMA szimbólumtartomány megegyezik 128 TDMA szimbólumtartománnyal. Egy *minislot* állítható számú továbbító kódot tartalmazhat, melyek száma 2 és 128 között lehet.

Tegyük fel, hogy *minislot*-onkénti kódok száma 4, és a kerethossz 16. Ekkor a *minislot* 64 szimbólumot tartalmaz, és az adott kód 16 S-CDMA szimbólumnyi ideig van egyazon felhasználóhoz rendelve. A példában leírt ugyanazon *minislot* által párhuzamosan továbbított 4 szimbólum kódosztásosan multiplexelt.

Ha egy S-CDMA keret minden *minislot*-ja ugyanazon modemhez van rendelve, az S-CDMA tiszta CDM-é válik azon időintervallum alatt. A másik szélsőséges helyzet akkor áll elő, ha a *minislot* csak két kódot tartalmaz, és minden *minislot* más kábelmodemhez van rendelve. Ekkor ezen keret alatt CDMA közeghozzáférés van 64 kábelmodem jel között.

CMTS	Lépés	CM
Sync, UCD és US MAP <i>minislot</i> információkat sugározása a DS csatornában.	1	1 DS csatorna keresése, az UCD-ért
		2 Szinkronizáció, megkapja az UCD-t
		3 RNG-REQ-et küld
RNG-RSP válasz = -1, 0, 1 teljesítmény állítási instrukcióval	4	
		5 Beállít és RNG-REQ újra
RNG-RSP válasz = -1, 0, 1 teljesítmény állítási instrukcióval	6	
		7 Beállít és RNG-REQ újra
RNG-RSP válasz = -1, 0, 1 teljesítmény állítási instrukcióval	8	
		9 Beállít és RNG-REQ újra
Szint jó, RNG-CMP Ideiglenes SID azonosító	10	
		11 SID vételt. A Raging kész!

14. ábra. Ranging

A 13. ábrán látható táblázat a modem távolságmérési folyamatát hivatott ábrázolni, melynek lépései a következők:

1. A *CMTS* az előreirányú csatornán keresztül folyamatosan küldi a szinkronizációs, a visszirányú csatorna leíró és a visszacsatorna időréseinek kiosztását tartalmazó *MAP* csomagokat.
- 2-3. A kábelmodem veszi az előbbi csomagokat, és az *UCD*-ben meghatározott frekvencián, sáv szélességben és modulációval, a *MAP* által engedélyezett időrésekben elküldi a *CMTS*-nek a MAC címmel együtt a fizikai kapcsolat felépítésének kérését (*RNG-REQ*).
4. A *CMTS* regisztrálja a kérését és a válaszában (*RNG-RSP*) utasítást ad a modem kezdeti adási szintjének 1 dB-es növelésére, vagy csökkentésére.
5. A modem vett üzenetben szereplő beállítást elvégzi, majd újra küldi a *RNG-REQ* kérését. (A 3-5. lépések addig ismétlődnek, amíg a kábelmodem központi egység megfelelőnek nem találja az adási szintet. Ha ezt a modem nem tudja elérni, akkor a *CMTS* visszautasítja a kérését és újra kezdődik a folyamat.)
10. A *CMTS* „sikeres beállítás” üzenetet (*RNG-CMP*) küld a modemnek, jelezve, hogy most már megfelelő az adási szint. Elküld egy ideiglenes *SID* azonosítót.
11. A kábelmodem nyugtázza a sikeres *Ranging*-et.

Ezek után elindul a regisztrációs folyamat, melynek részleteivel a következő fejezet foglalkozik.

4.3. Hálózati réteg

Ahhoz, hogy a modemek adatkommunikációt tudjanak megvalósítani egymással, valamint a külvilággal, tudniuk kell, hogy hova küldjék az adatot, illetve, hogy kitől kapták azt. Erre a világon jelenleg mindenhol használt *IP* címzési rendszert használják. A kommunikáció lebonyolításához az eszközöknek rendelkezniük kell a saját címmel, valamint adatküldés esetén ismerniük kell a céleszköz *IP* címét. Az *IP* címek modemekhez, előfizetői

eszközökhöz (röviden *CPE*), valamint a *VoIP* telefonáláshoz szükséges adapterekhez, azaz az *MTA*-khoz rendelését természetesen egy vagy több *DHCP* szerver végzi.

Bár néhány *CMTS* maga is el tudja látni ezt a feladatot, mégis ezt a hatáskörén kívül szokták helyezni, és a *CMTS*-nek ekkor csak *DHCP relay* feladatot kell ellátnia, azaz továbbítania a kéréseket a megfelelő *DHCP* kiszolgálóhoz. A legtöbb *CMTS*-en lehetőség van eszköztípusonként különböző kiszolgálót definiálni, vagyis külön-külön egyet a *CM*, a *CPE*, és az *MTA* eszközök számára.

A kábelmodemek számára természetesen csak privát *IP* címeket szokás rendelni, hiszen semmilyen olyan szolgáltatást nem tudnak nyújtani, amely indokolná a publikus *IP* használatát. Az *MTA*-kra is ugyanaz vonatkozik, mint a kábelmodemekre. Esetükben sincs szükség a publikus *IP*-kre.

A *CPE*-khez célszerű több, 256-osnál nem nagyobb *IP* tartományt rendelni. Ennek a szeparálásnak a gyakorlati oka, hogy ha például egy végpont férgek vagy egyéb okok miatt *broadcast*-on elkezdené terhelni a hálózatot, akkor az ne érintse az egész szolgáltatási körzetünket. Arról, hogy publikus vagy privát címek kerülnek kiosztásra az előfizetői végpontnak természetesen a szolgáltatók belátásuk szerint dönthetnek. Ez utóbbi esetben persze a világháló felé *NAT*-olni kell az ügyfelektől jövő csomagokat.

A privát *IP*-jú címkiosztásnak vannak előnyei és hátrányai is:

- Szolgáltatói szempontból hatalmas előny, hogy ez esetben nem kell akkora publikus *IP* címtartományt venni, ezzel jelentős költségeket megspórolva.
- Ugyanakkor az átlag felhasználók számára is pozitívumokat tartalmaz ez a módszer, hiszen az internet felől jövő támadások, betörések nem érhetik el az ő számítógépeiket, eszközeiket. Ez persze megfelelően beállította tűzfalat feltételez a hálózatban.
- Hátrány azonban az, hogy az ügyfeleknél néhány felsőbb rétegbeli protokoll által nyújtott szolgáltatás (mint például az *FTP*) nem működik megfelelően.

Az alaphelyzetben privát *IP*-kkel operáló szolgáltatók általában külön havidíj ellenében szoktak az ügyfeleknek publikus *IP* címet kínálni.

Mivel egy kábelmodemhez egy switch segítségével egyszerre több eszköz is kapcsolódhat, ezért célszerű megszabni azt is, hogy modemenként hány *CPE IP* osztható ki, ezzel lekorlátozva az egyidejűleg az internetre csatlakoztatható eszközök számát. Ha ez a mennyiség alacsony, akkor célszerű a *CPE IP*-k *lease time*-jét nem túl magasra állítani, különben néhány végponton levő eszköz nem fog tudni újbóli kapcsolatot létesíteni.

Egy másik megoldás a modemre közvetlenül kapcsolható eszközök számának korlátozására, ha figyeljük az *IP*-t igénylő *MAC* címét, és ez alapján osztunk, vagy nem osztunk hálózati címet az adott eszköz számára. Néhány szolgáltató például ezt úgy oldja meg, hogy csak a náluk egy webes felület segítségével beregisztrált *MAC* című eszközök számára oszt megfelelő *IP*-t az internet eléréséhez.

4.4. Applikációs réteg

Amikor a *DHCP* szerver válaszol a kábelmodem *DHCP* kérésére, akkor a válaszában nem csupán egy *IP* címet ad bérbe, hanem egyúttal megküldi a *ToD* és a *TFTP* szerverek címeit is, valamint a modem által letöltendő konfigurációs fájl nevét.

4.4.1. A *ToD* szolgáltatás

Miután a kábelmodem kapott *IP* címet, a *ToD* szerverrel való kapcsolatfelvételi kísérlettel folytatja tovább a működését. Ezen szerver címét a *DHCP lease*-el együtt kapja meg a modem.

Általános tévhit, hogy a *ToD* szolgáltatás, melyet a kábelmodem megpróbál igénybe venni miközben feljelentkezik a hálózatra, megegyezik az *NTP*-vel. Ám valójában az *NTP* és a *ToD* szolgáltatások inkompatibilisek. A legtöbb kábelmodem nem tud kommunikálni *NTP* szerverrel.

A *DOCSIS RFI* specifikációjának eleget tevő modemek tovább folytatják a feljelentkezési folyamatot, még akkor is ha a *ToD* szerver nem elérhető. Azonban mindaddig, amíg nem sikerül neki kapcsolatot létesíteni a szerverrel, rendszeresen újra próbálja azt. A korai *DOCSIS 1.0 RFI* specifikáció még azt mondta ki, hogy a modem egészen addig nem kerülhet online állapotba, míg nem sikerül elérni a *ToD* szervert, ám ezt később megváltoztatták.

4.4.2. A *TFTP* protokoll

Ahhoz, hogy a modem online állapotúvá váljon, hiányzik még egy igen fontos lépés. Be kell állítani a modem működési paramétereit, az adatátvitel módját, illetve a kapcsolat sebességét. Mindezek a modem konfigurációs fájljában lehet beállítani. Minden modem, miután megkapta az *IP* kommunikációhoz szükséges adatokat, letölti a központból ezt a fájlt a *TFTP* protokoll használatával.

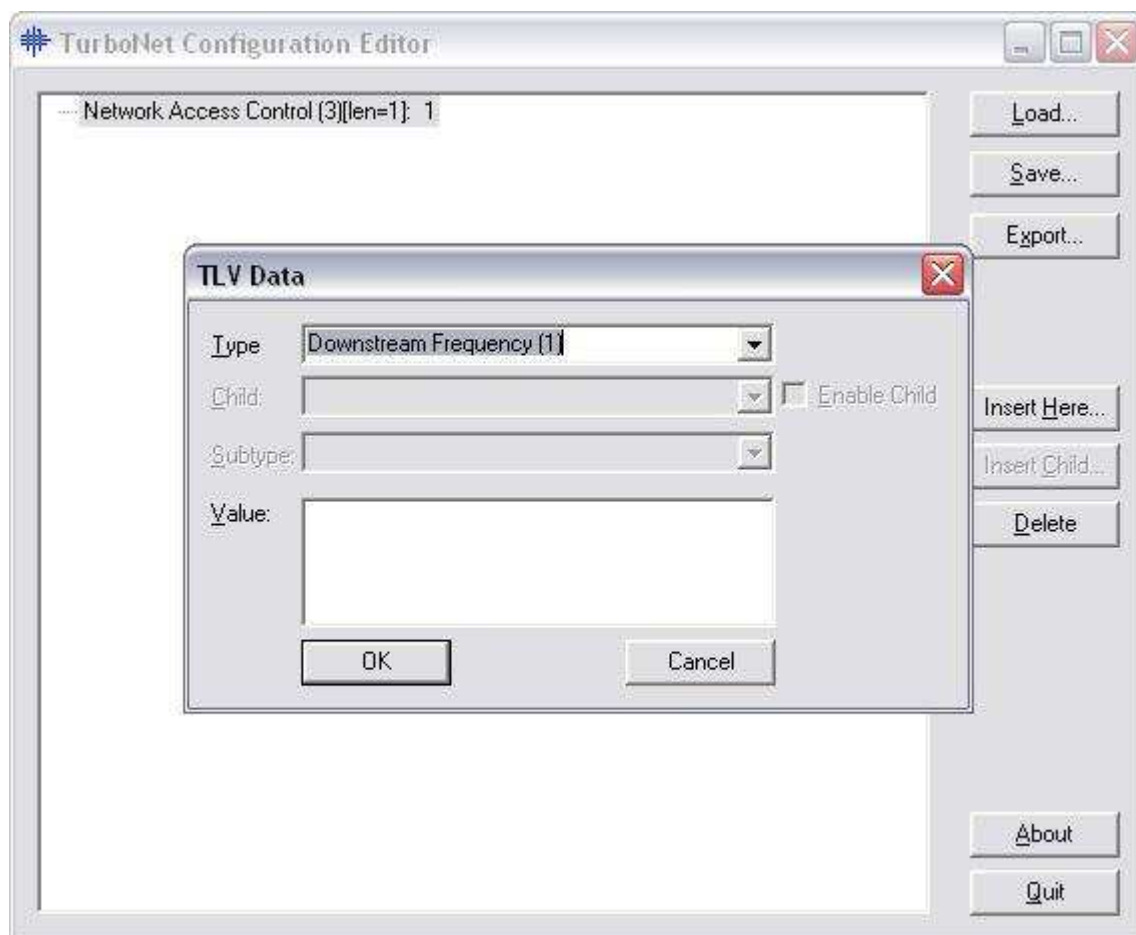
A protokoll, mint a nevéből adódik elég egyszerűen implementálható, és igen kevés erőforrást igényel. Könyvtárlistázásra nem képes, hanem mindössze fájlokat tud írni és olvasni a távoli szerveren. A 69-es *UDP* portot használja átvitelre. Legfeljebb 512 byte-os csomagok továbbításával zajlik az adatforgalom. Az átvitel „lock-step” módon megvalósított, azaz a kliensnek minden a szerverről történő olvasást vissza kell igazolni egy bizonyos időn belül, különben a csomag elveszettnek minősül. Csak a visszaigazolás megérkezése után történik meg az adat következő részének továbbítása, így biztosítva az adatok rendezettségét, szükségtelessé téve a csomagok későbbi újrendezését. A biztonság hiánya miatt csak privát hálózatokon használatos a protokoll. További részletek a [14]-es RFC dokumentumban.

A konfigurációs fájl emberi nyelven olvasható változatát írhatjuk kézzel is (hiszen egy egyszerű szöveges fájl), de vannak programok, melyekkel egyszerűen, kényelmesen lehet szerkeszteni azt. A 15-ös ábra mutatja, hogy az ingyenesen letölthető, TurboNet nevű programmal, hogyan lehet a konfigurációs fájl egy paraméterét beállítani.

Miután elkészült a fájl, generálni kell belőle egy a modemek által értelmezhető bináris állományt. Természetesen erre is létezik megfelelő segédprogram, mely képes szöveges konfigurációs fájlból binárisat, és binárisból emberi nyelvű fájlt előállítani.

Minden modemnek tudnia kell, hogy mi az általa a *TFTP* szerverről letöltendő konfigurációs fájl neve. Ezt a fájlnevet, ahogy már említettem, a *DHCP* válasszal egy időben megkapja a kábelmodem. Tehát a *DHCP* szervert kell megfelelőképpen beállítani, hogy el tudja látni ezt a feladatot. Talán a legtisztább megoldás, minden modemhez statikusan hozzárendelni egy *IP* címet és a konfigurációs fájljának a nevét. Ezt dinamikusan is meg lehetne tenni, ám erre nem minden *DHCP* szerver nyújt lehetőséget.

Lehetne díjsomagonként csak egy konfigurációs fájlt is létrehozni, a megfelelő sávzélességgel, ám sokkal célszerűbb megközelítés, ha minden modem egyéni konfigurációs fájlt kap. E módon, ha adott szükséges lenne lekorlátozni egy kábelmodemet, elég lenne csak számára új konfigurációs fájlt generálni, és felülírni azzal a korábbit.



15. ábra. A TurboNet konfigurációs fájl szerkesztő

Példaként álljon itt egy alapvető beállításokat tartalmazó konfigurációs fájl tartalma:

```
Main {  
    NetworkAccess      1;  
    DownstreamFrequency 417000000;  
    UpstreamChannelId  1;  
    MaxCPE             4;
```

```

ClassOfService {
    ClassID          1;
    MaxRateDown     1024000;
    MaxRateUp       256000;
    PriorityUp       3;
    GuaranteedUp    32000;
    MaxBurstUp      0;
    PrivacyEnable   0;
}
SwUpgradeFilename "DCM425-ST52.05.10-060616-S-D.img";
}

```

Ha a konfigurációs fájlban szerepel az opcionális `SwUpgradeFilename` paraméter, a modem megpróbálja letölteni az itt megadott, *firmware*-t tartalmazó image fájlt és frissíteni magát.

4.4.3. A hálózat monitorozása

Ahhoz, hogy a már működő rendszert, hatékonyan üzemeltetni lehessen, szükséges az kulcsfontosságú eszközök nyomon követése, monitorozása. Egy üzletszerűen használni kívánt hálózatnál elengedhetetlen egy jó provisioning rendszer megléte. Ennek segítségével bármikor átfogó képet lehet kapni a hálózat állapotáról, listát a hálózaton jelenleg fenn lévő modemekről, illetve le lehet kérdezni azok aktuális technikai adatait, műszaki paramétereit. Hálózathiba esetén a modemekről kapott listából, ismerve a hálózat fizikai felépítését, hamar meg lehet találni a hiba helyét, és annak valószínűsíthető okát.

Több cég is kifejlesztett ilyen provisioning rendszert és kínálja azt általában havidíj ellenében. Volt szerencsém több rendszerrel is megismerkedni, melyek közt igen jelentős tudásbeli eltérések lehetnek. Bár olykor igen jelentős összegeket kérnek egy komolyabb tudással rendelkező rendszer üzemeltetéséért, mégsem szabad ezen a ponton elspórolni a hálózatba fektetett pénzt, hiszen jelentős segítséget nyújtanak a mindennapi munka során.

Ezen rendszerek egyes funkciói az *SNMP* hálózatmenedzsment protokollon alapulnak.

4.4.3.1. Az *SNMP* protokoll

Az *SNMP* első verziója 1990-ben jelent meg, az *SGMP* utódjaként, majd nem sokkal később, '93-ban debütált az *SNMPv2*, és 2000-ben az *SNMPv3*. Az utóbbi verziók a kezdeti filozófiától nem térnek el, ugyanazon az elven működnek, csupán kibővítik annak funkcionalitását, valamint biztonságosabbá teszik azt.

A protokoll egyszerűsége abban rejlik, hogy nem igényel összeköttetés alapú kommunikációt. Ennek köszönhetően elegendő az *UDP* használata adattovábbításra.

Az *SNMP*-t alapvetően három összetevő alkotja:

- A felügyeleti komponensek
- A kiszolgáló program (agent, ügynök)
- *MIB*

Az *SNMP* által menedzselt csomópont bármilyen eszköz lehet (switch, router, gateway, stb.), amely képes az *SNMP* használatával információt gyűjteni és küldeni saját magáról. Természetesen ezeknek az eszközöknek képeseknek kell lenniük az *SNMP* ügynökök futtatására. Ezen ügynökök feladata, a felügyeleti komponensek kéréseinek kiszolgálása.

A felügyeleti komponensek a felügyeleti állomáson futnak. Feladatuk, hogy kéréseket küldjenek az ügynököknek, és fogadják, értelmezzék, megjelenítsék a tőlük érkező válaszokat.

Az *SNMP* protokoll önmaga nem határozza meg, hogy a menedzselt eszköznek milyen információkat nyújtanak, hanem csak azt, hogy milyen formában kell azokat rendelkezésre bocsátani. Ezen információk a *MIB*-ben vannak definiálva, amely egy virtuális adatbázis. Ezt kérdezik le, illetve módosítják a kéréseknek megfelelően az *SNMP* ügynökök.

Az adatbázis változókat, objektumokat tartalmaz strukturált, fa szerkezetben. Ezekre az objektumokra számláncokkal, vagy ezt helyettesítő nevekkel hivatkozhatunk (például 1.3.6.1.2.1.1.3 = iso.org.dod.internet.mgmt.mib-2.system.sysUpTime, röviden sysUpTime).

Fontos megjegyezni, hogy az *SNMP* objektum fogalma, nem ekvivalens az objektumorientált rendszerek objektum fogalmával.

Ezeket az objektumokat egy szabványos definíciós nyelv segítségével írják le, melyet *ASN. 1*-nek neveznek. Az *SNMP* nem használ minden az *ASN. 1*-ben definiált adattípust, ugyanakkor kiegészíti azt néhány új definícióval.

Az *SNMP* által használt üzenetek formátuma a következő:

Version number	Community String	SNMP PDU
----------------	------------------	----------

SNMP fejléc

A Community String rész az autentikációt szolgálja.

Az *SNMP* öt *PDU*-t definiál:

- **GetRequest:** A lekérdező applikáció generálja, majd küldi el az ügynöknek. Az ügynök kikeresi a *GetRequest PDU* „variable-bindings” mezőjében megadott *MIB* változót, és a keresés eredményét egy *GetResponse PDU* segítségével visszaküldi.
- **GetNextRequest:** Szintén a lekérdező alkalmazás generálja, és küldi el az ügynök számára. Az ügynök a „variable-bindings”-ban hivatkozott azonosítót lexikailag követő *MIB* változót keresi ki.
- **SetRequest:** Ugyanúgy a lekérdező generálja, és küldi. Az ügynök kikeresi a „variable-bindings”-ban megadott *MIB* változót és annak értékét beállítja a *SetRequest PDU* által tartalmazott értékre. Majd válaszként visszaküld egy *GetResponse PDU*-t.
- **GetResponse:** Az ügynök a kapott kérésre adott válaszát tartalmazza. A „request-id” megegyezik a kérés „request-id”-jével. Ha a kérés feldolgozása során (keresés, érték beállítás) hiba lépett fel, akkor az „error-status” és az „error-index” mezők tartalmazzák a hiba leírását (egyébként értékük 0). Amennyiben nem történt hiba, a *GetRequest* és a *GetNextRequest* kérésekre adott válasz tartalmazza a kért változó értékét is.
- **Trap:** Olyan speciális *PDU*, amelyet az agent automatikusan generál és küld el a menedzser komponensnek, amikor egy meghatározott állapotot érzékel.

Ilyen állapotok:

- **ColdStart:** hideg újraindítás (ekkor a konfiguráció módosulhat)
- **WarmStart:** meleg újraindítás (a konfiguráció nem módosulhat)
- **LinkDown:** hiba történt az egyik kommunikációs kapcsolattal
- **LinkUp:** a kapcsolat helyreállt

- AuthenticationFailure: jogosulatlan hozzáférés
- egpNeighbourLoss: forgalomirányító hibajelzése
- EnterpriseSpecific: valamilyen gyártófüggő esemény bekövetkezése esetén

Az *SNMPv2*-től kezdve két további *PDU* is van a protokollban:

- GetBulkRequest: A táblázatok gyorsabb lekérdezésére szolgáló iterátor.
- InformRequest: Ez a *PDU* lehetővé teszi a felügyeleti állomás számára, hogy egy másik felügyelőt informáljon a számon tartott változókról.

A GetRequest, a GetNextRequest, a SetRequest, és a GetResponse *PDU*-k felépítése:

PDU type	request-id	error-status	error-index	variable bindings
----------	------------	--------------	-------------	-------------------

A Trap *PDU* felépítése az *SNMP* első verziójában, ahogy azt a következő táblázat mutatja még ettől különböző volt, de az *SNMPv2*-től kezdve már megegyezett a többiével.

PDU type	Enterprise	Agent address	Generic trap type	Specific trap type	Timestamp	Variable bindings
----------	------------	---------------	-------------------	--------------------	-----------	-------------------

Az *SNMP* ügynökök a 161-es *UDP* porton fogadják a kéréseket (bármely portról érkezhetsz a kérés, az ügynök a forrás porton fog válaszolni), a Trap üzenetek számára pedig a 162-es port van fenntartva.

Összefoglalva az *SNMP*-ről azt lehet elmondani, hogy egyszerűsége révén egy könnyen implementálható és jól alkalmazható hálózatmenedzsmenti szabvány. Azonban biztonsági lehetőségei (főleg az első két verzióban) igen korlátozottak, a nagy tömegű adatlekérdezést nem támogatja (az első verzió), valamint az összeköttetés-mentes kapcsolat miatt óvatosan kell alkalmazni nagy biztonságot igénylő rendszerek esetén.

4.4.3.2. A Nagios rendszer

Ahhoz, hogy a felépített rendszerünk később esetlegesen bekövetkező hibáit hamar el tudjuk hártani szükséges, hogy arról mielőbb tudomást szerezzünk. Ebben nyújt igen hatékonyan segítséget a nyílt forráskódú Nagios hoszt- és szolgáltatásmonitorozó rendszer. Futtatásához valamilyen UNIX alapú operációs rendszer szükséges, valamint a webes felület használatához egy tetszőleges, *CGI* futtatására képes webserverver.

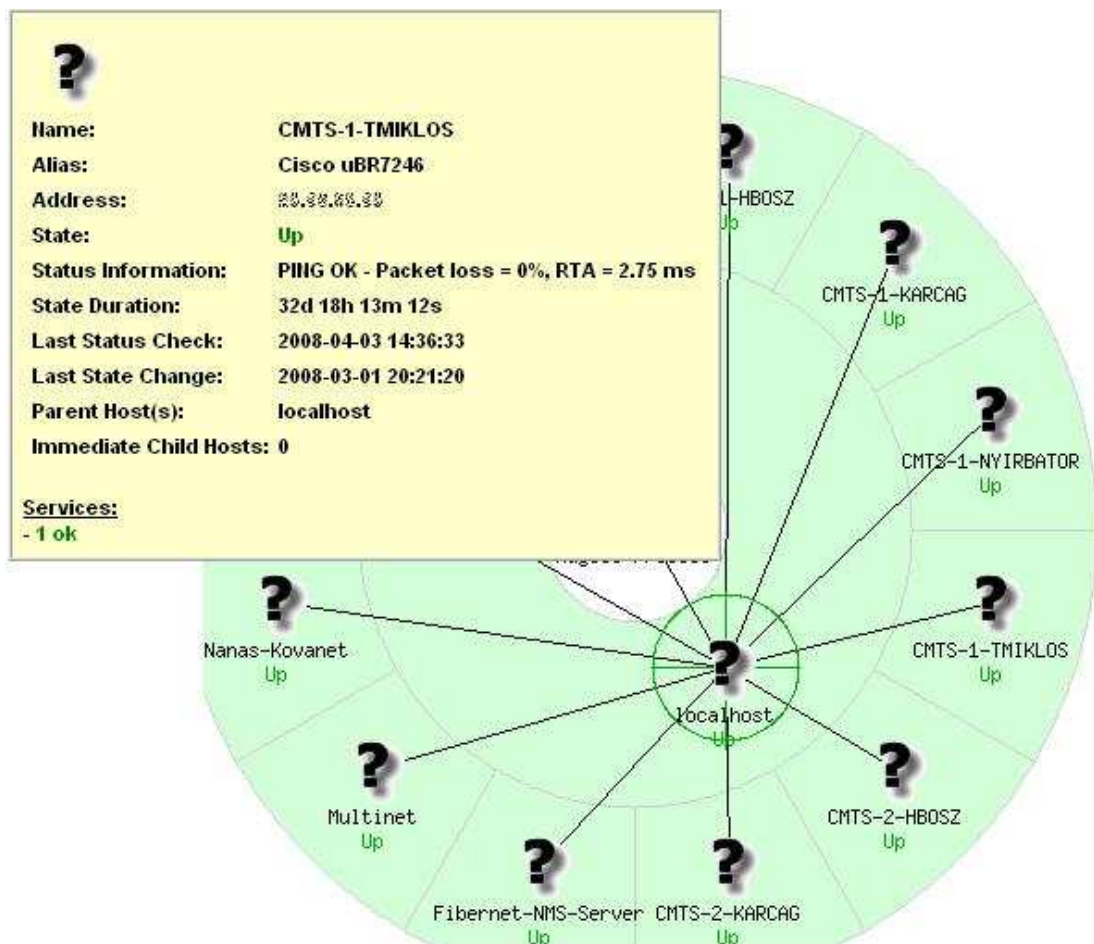
A Nagios elsődleges feladata az általunk beállított hálózati eszközök (routerek, *CMTS*-ek, stb.) megfelelő üzemelésének ellenőrzése, azaz szolgáltatásainak, erőforrásainak (processzor terheltség, lemez- és memóriahasználat, futó folyamatok, stb.) figyelése. Ezen túlmenően képes (természetesen megfelelő szenzor megléte esetén) a külső hőmérséklet figyelésére is. Bizonyos hibák érzékelése esetén képes akár beavatkozni is, megpróbálva elhárítani azt az általunk előre megadott módon (például a leállt szolgáltatás újraindításával).

Számunkra a leghasznosabb (és az egész rendszer elsődleges) funkciója, hogy képes figyelni egy hoszt elérhetőségét. Ezt bizonyos időközönkénti pingeléssel teszi. Ha a sikertelen ellenőrzési kísérletek száma elér egy adott szintet, akkor figyelmeztetést küld a beállított csoportnak. Ez az értesítés történhet e-mailen, személyhívón, MSNen, vagy egyéb a felhasználó által beállított módon. A legjobb megoldás talán, ha a mobiltelefon szolgáltatónknál regisztrálunk egy e-mail címet, és azt adjuk meg értesítési címmek. Ez azzal az előnnyel jár, hogy a szolgáltatóknál van lehetőség SMS értesítőt kérni e-mail érkezése esetén, mely akár a levél egy részét is tartalmazhatja. Így vonalhiba, vagy leállás esetén rögtön tudni fogjuk, hogy mikor és hol történt az. Természetesen akkor is képes értesíteni minket a rendszer, ha az adott végpont újra elérhetővé vált.

Az előbbi paramétereket (hoszt címe, ellenőrzési időköz, ellenőrzések száma figyelmeztetés küldése előtt, értesítési cím, stb.) mind mi adjuk meg a rendszernek, amely arra is figyel, hogy több hoszt monitorozása esetén ne egyszerre vizsgálja őket, csökkentve így a hálózati terheltséget. A hosztok között hierarchiát is definiálhatunk, jelezve így az eszközök egymással való kapcsolatát. Ezt figyelembe véve a Nagios, ha egy végpont nem válaszol, ellenőrzi, hogy a hozzá vezető további állomások elérhetők-e, így megkeresve a hiba valószínűsíthető forrását.

Az előbbi beállításokon túl, lehetőség van a vizsgálatok számára időintervallumot beállítani, hogy csak akkor figyelje az adott eszközt. Továbbá előre betervezett leállások felvételére is van mód, ha például egy szolgáltatást újraindítani, esetleg frissíteni szeretnénk.

A beállítások mindegyikét, a telepítés után konfigurációs fájlok írásával lehet megtenni. Miután módosítottunk a beállításokon, újra kell indítanunk a Nagios azok életbe léptetéséhez.



16. ábra. A Nagios szolgáltatástérképe

A Nagios rendszer által nyújtott előnyök, szolgáltatások, összefoglalva:

- Hálózati eszközök, szolgáltatások, erőforrások monitorozása
- Lehetőség a hosztok közti hierarchia definiálására, ezáltal lehetővé téve a leállt, és nem elérhető végpontok felismerését és megkülönböztetését
- A megadott csoportok értesítése hiba észlelése és megszűnése esetén
- Eseménykezelők létrehozásának lehetősége problémák automatikus elhárítására

- Támogatás redundáns monitoring szerver létrehozásához
- Webes felület a hálózat jelenlegi állapotának, a múltbéli figyelmeztetések és problémák, a log fájlok, stb. megtekintéséhez
- A monitorozó és figyelmeztető viselkedés menet közbeni módosításának lehetősége eseménykezelőkön, webes felületen, és külső alkalmazásokon keresztül
- A webes interfész használatának bizonyos felhasználókra való korlátozása

4.4.3.3. MRTG

További hasznos információkhoz, statisztikákhoz, grafikonokhoz juthatunk az MRTG, azaz a Multi Router Traffic Grapher használatával. Ennek elsődleges funkciója a hálózati eszközökből nyert adatátviteli információk grafikonos formában való megjelenítése. Ezen túlmenően képes még a különféle erőforrások kihasználtságának hasonló formájú megjelenítésére is.

Az MRTG egy Perl scriptből és egy C programból áll. A Perl script feladata, hogy *SNMP* használatával forgalominformációt gyűjtsön a routerektől, és más hálózati eszközöktől. Ezeket az információkat logolja és használja fel a C program a hálózati forgalmat ábrázoló grafikonok generálásakor, melyek *PNG* formátumban kerülnek mentésre. Ezen grafikonok jelennek meg a bármely modern böngészővel megjeleníthető weboldalon.

Az MRTG a korai időszakban pusztán Perl alapokra támaszkodott, azonban ez nem volt túl hatékony, igen lassúnak bizonyult. Ezért került sor a program időkritikus részének C belé implementálására.

A részletes napi nézetén túlmenően, az MRTG további grafikonokat is generál az utolsó hét nap, az utolsó öt hét, és az utolsó tizenkét hónap forgalma alapján. Ezt a log fájlok teszik lehetővé, ugyanis az MRTG naplóz minden a routertől lekért adatot. A log két évre visszamenően tartalmaznak minden releváns adatot.

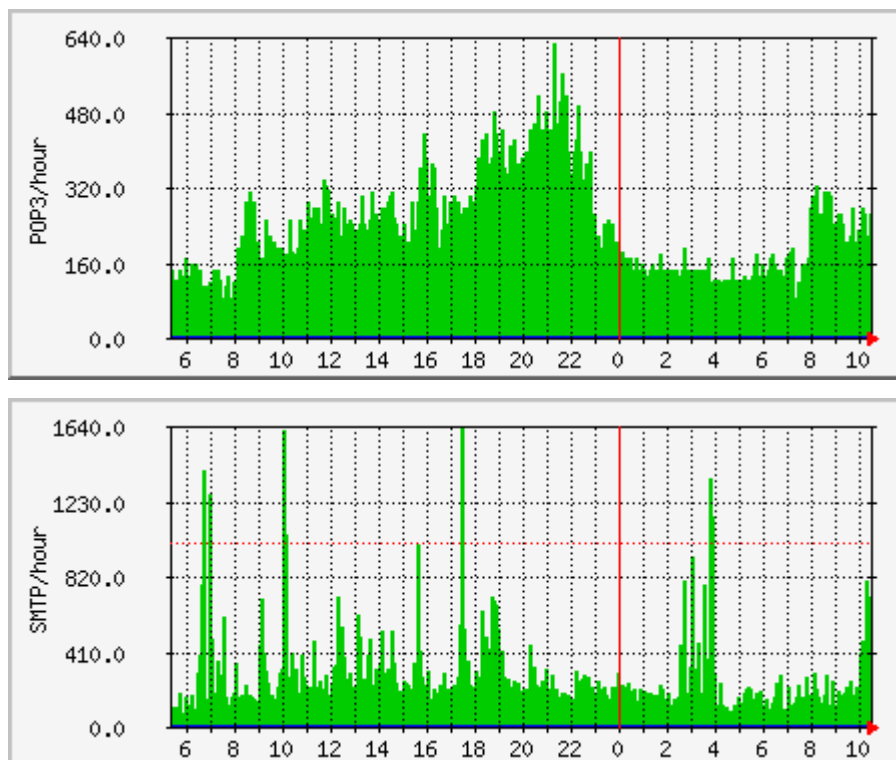
Az MRTG funkcionalitása nem merül ki csupán a forgalom nyomon követésében. Lehetőség van bármely *SNMP* változó monitorozására, mint például a rendszer erőforrásainak

kihasználtsága. Továbbá lehetséges két vagy több forrás adatainak megjelenítésére egy grafikonon.

Az MRTG használatához valamilyen Windows NT, vagy UNIX alapú platform, valamint egy webszerver szükséges. Rendelkezésre áll egy cfgmaker nevű, konfigurációs fájlt generáló parancssoros segédprogram, melynek használatával könnyen megadhatjuk a működéshez szükséges paramétereket. A program képes kilistázni egy adott eszközből kinyerhető és monitorozható változókat. Ezen változók megadásával definiálhatjuk, hogy egy eszköztől mely információk legyenek megjelenítve az MRTG által.

Az MRTG tulajdonságai, előnyei:

- Használható mind Windows NT, mind UNIX platformon
- Nyílt forráskódú, így van mód a módosításra, továbbfejlesztésre
- Állandó méretű log fájlok, melyek mérete a tömörítő algoritmusnak köszönhetően nem növekszik
- Könnyű konfigurálhatóság a hozzá való segédprogrammal
- Az MRTG webes kinézete könnyen változtatható



17. ábra. Egy levelezőszerver két MRTG grafikonja

5. A hangszolgáltatás eszközei

Mivel ez a fejezet egy különálló írást is megérvő témát boncolgat, ezért itt csupán a *DOCSIS* technológiához szorosan kötődő PacketCable architektúráról, a kommunikációban magának egyre nagyobb teret hódító *SIP* protokollról, és a *VoIP* szolgáltatók körében igen népszerű Asterisk *PBX*-ről esik bővebben szó. Ezen alapvető elemek megismerésével képet kaphatunk a *VoIP* szolgáltatás alapjairól.

5.1. A PacketCable architektúra

A *DOCSIS* megjelenése után nem sokkal, 1997-ben a CableLabs tagjai felismerték az igényt egy valósidejű multimédia architektúra megalkotására, amely lehetőséget ad fejlett szolgáltatások nyújtására a *DOCSIS* kábelmodem rendszereken keresztül. Ennek a kezdeményezésnek lett az eredménye a PacketCable.

A PacketCable egy olyan *IP* alapú architektúrát határoz meg, mely a *DOCSIS* 1.1-es (vagy magasabb) változatára épül. Bár a *VoIP* szolgáltatás biztosítása volt az elsődleges megfontolás a platform létrehozására, az a hangátvitelen túl támogat további szolgáltatásokat is, mint például a videokonferenciák, vagy az interaktív játékok.

A végpont-végpont architektúra egy komplett rendszert kínál, mely magába foglalja a provisioning-et, a konfiguráció menedzsmentet, a *QoS*-t, valamint a jeladást, az eseményüzeneteket, és a biztonság kezelését. Ezen funkciók kezelését különálló szerverek és hálózati végpontok végzik, melyek együttesen alkotják a PacketCable hálózatot.

A fejállomás nyílt és szabványos interfészek segítségével együttműködő szerverei szolgálnak a PacketCable hálózat magjaként. A *DOCSIS* 1.1 infrastruktúráján alapuló szerverek fogják össze a *HFC* hálózat PacketCable előfizetőit; a menedzselt *IP* hálózatot; és azokat az átjárókat, melyek összekapcsolják a *PSTN* törzseket az *SS7* hálózattal.

A PacketCable architektúrájának áttekintése

A *CMTS* a *DOCSIS* protokollt használja a kábelmodemek vezérlésére az osztott *HFC* hálózaton. A multimédia terminál adapter, a kábelmodemek PacketCable kiegészítője nyújtja

az *IP*-re épülő szolgáltatásokat. Ezek az eszközök lehetnek különállóak, vagy lehetnek a kábelmodembe ágyazottak is. A menedzselt *IP* hálózat teremti meg a kapcsolatot a jeladásért, a médiáért, a provisioningért, és a *QoS*-ért felelős, alapvető funkciójú komponensek között. Továbbá, a menedzselt *IP* hálózat hosszú távú *IP* kapcsolatot biztosít más menedzselt *IP* és *DOCSIS HFC* hálózatok, illetve a *PSTN* átjárók között.

A *DOCSIS* 1.1-nek két fő előnye van erre az architektúrára nézve, az 1.0-ás verzióhoz képest: egyrészt támogatást biztosít a *QoS* számára, másrészt továbbfejlesztett biztonságú a kábelmodemek autentikációja a hálózaton, biztonságosabb a modemek szoftverletöltése, és biztosított a *CMTS* és a modemek „privát” kommunikációja.

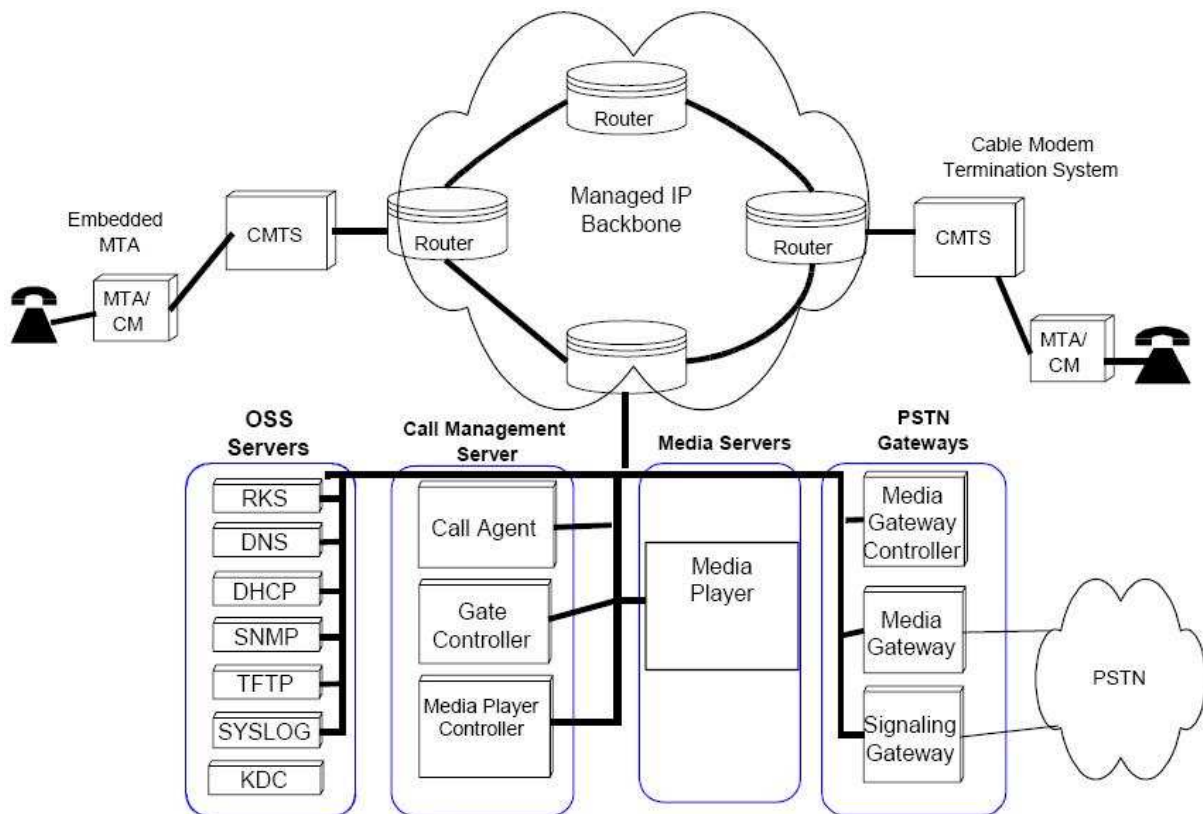
A hálózat felépítése

Ahogy az már az előzőekből kiderült a PacketCable szolgáltatások nyújtásához az ügyfelek számára legalább *DOCSIS* 1.1 kompatibilis, PacketCable-re felkészített modemeket kell biztosítanunk, melyek tartalmazzák az *MTA*-t (vagy különálló *MTA*-t is elhelyezünk náluk). A *CMTS*-nek szintén legalább *DOCSIS* 1.1 kompatibilisnek kell lennie a lépcsőzetes szolgáltatások és a biztonságosabb hozzáférésű hálózat miatt.

A PacketCable lehetővé teszi az *MTA*-k számára, hogy átjárókon keresztül együttműködjenek a *PSTN*-el. Ez a *PSTN*-el való kapcsolat a *Media Gateway Controller (MGC)*, a *Media Gateway (MG)*, és a *Signaling Gateway (SG)* kombinációja. Az *SG* cseréli ki az *ISUP* és a *TCAP* üzeneteket a *PSTN SS7* hálózatával.

Van egy hibrid megoldás is, amely PacketCable-t használ a hálózat hozzáférési részén, valamint a *PSTN* helyi digitális kapcsolója általi vonalkapcsolt hívásvezérlést. A csomag- és vonalkapcsolt kommunikáció közötti váltás az *IPDT*-ben történik meg. Ezen megoldás előnye, hogy hidat képez a teljes *IP* alapú hálózathoz, lehetővé téve, hogy a *PSTN* kapcsoló végezze a provisioning, a számlázási, az operátori, és a sürgősségi feladatok nagy részét.

Az alábbi 18. ábra egy teljes funkcionalitású PacketCable hálózatot ábrázol.



18. ábra. A teljes PacketCable hálózat

Call Management Server: A CMS a kulcskomponens, amely a hívásvezérlési és jeladási szolgáltatásokat nyújtja a PacketCable hálózat MTA-i, CMTS-ei, és PSTN átjárói számára. A CMS két főbb összetevőből, egy Call Agent-ből és egy Gate Controller-ből áll. A Call Agent utasítja az MTA-t, hogy figyeljen bizonyos eseményeket, vagy hogy játsszon le konkrét jelzéseket, mint például a tárcsahang. A Gateway Controller komponens QoS engedélyezési feladatokat lát el. Ez kommunikál a CMTS-el, hogy engedélyezze, vagy elutasítsa az MTA QoS iránti kérelmét.

Operational Support Systems: A működtető rendszerek különböző támogató szerverek és infrastrukturális funkciót tartalmaznak, melyek támogatást nyújtanak a szolgáltatás üzemeltetéséhez. Ilyen például a Record Keeping Server, amely PacketCable eseményeket fogad a CMS-től, a CMTS-től, és a Gateway Controller-től. Ez összegyűjti a esemény-üzeneteket egy koherens halmazba, vagy CDR-be, amely aztán elérhető egyéb háttérirrodai rendszer számára, mint például a számlázó, vagy a csalódetektáló.

Media Servers: A média szerver (másképpen audió szerver) tárolja a média tartalmat egy sajátos multimédiaszolgáltatás számára. A PacketCable-ben a média szerver továbbítja az előfizetőnek az olyan üzeneteket, mint például: „az előfizető átmenetileg nem kapcsolható...”. A Media Player Controller kéri a Media Player-t, hogy játssza le ezeket a hálózati közleményeket, melyek a CMS által meghatározott hívásállapottól függenek.

5.2. SIP

A *SIP*, azaz a *Session Initiation Protocol* egy az ISO-OSI modell viszony rétegében elhelyezkedő jeladó protokoll, mely lehetővé teszi több felhasználós kommunikációs *session* kialakítását, függetlenül az átvinni kívánt médiatartalomtól, legyen az hang-, videó-, adat- vagy Web-alapú. Számos szoftver és eszköz van a piacon, melyek támogatják a *SIP* szolgáltatásait, köztük *IP* telefonok, *VoIP* átjárók, média szerverek, satöbbi. Elterjedtségét talán mi sem bizonyítja jobban, minthogy a 3G közösség a *SIP*-et választotta az új generációs mobiltelefon hálózat *session* vezérlő mechanizmusának. Ezen túlmenően, sok más terület mellett, az MSN hálózat is ezt a protokollt használja.

A nagy múltú *SS7* szabvánnyal és a *H.323* videó protokollkészlettel ellentétben, a *SIP* az alatta levő hálózati protokolloktól és adatátviteli médiumoktól függetlenül működik. Ellenben, ez a protokoll csak azt határozza meg, hogy a kommunikációban résztvevő eszközök hogyan, milyen módon hozhatják létre, módosíthatják, bonthatják a kapcsolatot.

A *SIP* jelentős előrelépés olyan protokollokhoz képest, mint például a Media Gateway Control Protocol, mely a *PSTN* audió jeleit *IP* adatcsomagokká alakítja. Mivel az MGCP zárt, csak-hang szabvány, ezért ennek továbbfejlesztése jelző képességekkel bonyolult, valamint esetleg hibás és eldobott üzeneteket eredményezhet, így akadályozva a szolgáltatókat új szolgáltatások kínálásában. A *SIP*-et használva viszont a programozók új információkat adhatnak az üzenetekhez a kapcsolat veszélyeztetése nélkül.

A *SIP* jelentősen hasonlít a *HTTP* és az *SMTP* protokollokra, a *session*-öket hozzájuk hasonlóan építi ki, hiszen javarészt az Internet alapvető szolgáltatásainak a működését vették alapul a tervezésekor. Ez azzal az előnnyel szolgál, hogy a programozók ezáltal könnyen és

gyorsan írhatnak *SIP*-re alapuló alkalmazásokat, jelentősen lerövidítve így az új kommunikációs szolgáltatások kifejlesztésének idejét.

Egy *SIP* üzenet nagyon hasonlít a HTTP-hez. Számos üzenetfejléc és HTTP kód újrafelhasználásra került. Például a cím nem elérhető hiba kódja a *SIP*-ben szintén 404-es. A *SIP* az SMTP-t is újrafelhasználja a címzési sémáknál. Egy *SIP* cím, mint például sip:guest@example.com, struktúrája tökéletesen megegyezik egy email cím struktúrájával, és ugyanazon megszorítások is érvényesek rá.

A címek használatához természetesen a DNS bejegyzések közé újakat kell felvenni. Ennek megvalósíthatósága érdekében két új bejegyzés típus jelent meg: az *SRV* és az *NAPTR*.

Mivel a *SIP* nem egy session leíró protokoll, és nem nyújt konferenciavezérlést sem, ezért az *SDP*-t használja az üzenetek tartalmának és karakterisztikájának leírására. A *SIP* maga *QoS*-t sem nyújt, ezért az *RSVP*-vel működik együtt a hangminőség biztosítása érdekében. Az előbbieken kívül számos további protokollt használva működik, többek között az LDAP-t a tartomány meghatározására, a *RADIUS*-t az autentikációra és az *RTP*-t a valós idejű átvitelhez.

A SIP főbb tulajdonságai:

- Nem határozza meg a létrehozandó kapcsolatot, csak azt, hogy hogyan kell azt menedzselni. Ez nagyfokú rugalmasságot biztosít, így a *SIP* rengeteg alkalmazásban hasznosítható.
- A *SIP* üzenetek szövegesek, így könnyen olvashatóak és debuggolhatóak, megkönnyítve a fejlesztést.
- A *SIP* az email kliensekhez hasonlóan felhasználja a *MIME típus* leírásokat, így a *session*-ökhöz társított alkalmazások automatikusan elindulhatnak.
- A *SIP* felhasználja a már létező, elterjedt internet szolgáltatásokat, mint például a DNS, az *RPT*, vagy az *RSVP*. A legtöbb ilyen szolgáltatás helyben már megtalálható, vagy könnyen elérhető.
- A *SIP* bővítmények a hálózat károsítása nélkül hozzáadhatók az új alkalmazásokhoz. A hálózat régebbi *SIP* alapú eszközei nem korlátozzák az újabb *SIP* szolgáltatásokat.

Ha egy régebbi *SIP* implementáció nem támogat egy újabb *SIP* alapú alkalmazás által használt eljárást vagy fejléct, akkor az egyszerűen figyelmen kívül hagyja.

- A *SIP* transzport réteg független. Működhet mind *UDP*, mind *TCP* protokollon keresztül, rugalmasan összekötve a felhasználókat az alapszintű infrastruktúrától függetlenül.
- Ha egy alkalmazás, vagy *session* videó- és hangátvitelt kezdeményez, a hang még akkor is továbbítható egy eszköznek, ha az nem rendelkezik videótámogatással.

A négy fő összetevő, melyet a *SIP session* használ:

- *SIP User Agents*: Ezek a végfelhasználói eszközök (mobiltelefonok, PCk, PDAk, stb.) hozzák létre és menedzselik a *SIP session*-t. A User Agent kliens elindít egy üzenetet, melyre a User Agent szerver válaszol.
- *SIP Registrar Servers*: Adatbázisok, melyek a User Agent-ek helyét tartalmazzák a tartományon belül. A *SIP* üzenetváltás során ezek a szerverek fogadják és küldik tovább a résztvevők *IP* címét és egyéb vonatkozó információkat a *SIP Proxy Server*-nek.
- *SIP Proxy Server*: Fogadja a *SIP User Agent*-ek *session* kéréseit, és lekérdezi a *SIP Registrar Server*-t a kért User Agent címzési információért. Ezt követően továbbítja a *session* meghívást közvetlenül az illető User Agent-nek, ha ugyanabban a tartományban vannak, illetve egy másik Proxy Server-nek, ha a User Agent nem az adott tartományban van.
- *SIP Redirect Server*: Lehetővé teszi a *SIP Proxy Server*-ek számára, hogy *SIP session* meghívást küldjön külső tartományokba. A *SIP Redirect Server* természetesen lehet a *SIP Registrar Server*-ekkel és a *SIP Proxy Server*-ekkel azonos hardveren.

Az RFC3261-ben definiált *SIP* metódusok:

- *INVITE*: Egy másik User Agent a *session*-be való meghívására szolgál.
- *RE-INVITE*: Az éppen futó *session* módosítására használható.
- *REGISTER*: Egy hely a Registrar Server-ben való regisztrálására szolgáló kérés.
- *ACK*: Visszajelzés az *INVITE* sikeres megérkezéséről és elfogadásáról.
- *CANCEL*: Egy elküldött *INVITE* kérés visszavonására való.
- *OPTIONS*: Más User Agent vagy Proxy Server képességeinek lekérdezésére szolgál.

A kérések sikerességének vagy sikertelenségének jelzésére a *SIP* Response-ok, azaz válaszok hivatottak, melyeket valójában a HTTP-hez hasonlóan egy kód szimbolizál.

5.3. Az Asterisk *PBX*

Az Asterisk a Digium cég nyílt forráskódú, ingyenes *PBX* implementációja. Mint minden egyéb *PBX*, lehetővé teszi a hozzá kapcsolódó telefonkészülékek, szoftverek számára, hogy hívásokat kezdeményezzenek és bonyolítsanak le egymással, valamint hogy más telefonszolgáltatásokhoz kapcsolódjanak (beleértve a *PSTN*-t is). A szoftver alapvetően UNIX alapú rendszerekre lett fejlesztve (C nyelven), de elérhető már a Windows-os környezetre átírt változata is.

Az alap Asterisk szoftver számos a védett *PBX* rendszerekben is megtalálható lehetőséget nyújt, mint például a hangposta, a konferenciahívás, vagy az automatikus hívás elosztás. A beépítettekén kívül van lehetőség saját funkciók definiálására is hívásterv szkriptek írásával az Asterisk saját nyelvén, egyéni C modulok beépítésével, vagy Perl nyelvű *AGI* szkriptek írásával.

Az Asterisk számos jeladó protokollt, köztük a *SIP*-t, az *MGCP*-t, és a *H.323*-at, valamint média átviteli protokollt támogat, mint például az *RTP*. Együtt tud működni a legtöbb *SIP* telefontal, illetve képes registrar-ként, valamint az *IP* hálózat és a *PSTN* közti átjáróként is egyaránt üzemelni. Az Asterisk fejlesztői egy új protokollt is kidolgoztak a hívások Asterisk *PBX*-ek, és más ezen protokollt támogató *VoIP* szolgáltatást nyújtó eszközök közti hatékony továbbításához. Ezt Inter-Asterisk eXchange-nek nevezték el, melynek jelenleg az IAX2-es változata van elterjedve. Néhány telefonkészülék az Asterisk szerverekkel való kommunikációhoz közvetlenül támogatja az IAX2-t. A telefonkészülékek mellett több egyéb, az Asterisk-el együttműködni képes eszköz is megjelent a piacon, mind a Digium, mind külső cégek tervezésében.

Az Asterisk helyes működéséhez két dolgot kell beállítanunk: a hangátviteli csatornákat és a tárcsázási tervet. Ezeket, mint általában a szerverprogramok esetében, a megfelelő konfigurációs fájl megírásával/módosításával tehetjük meg.

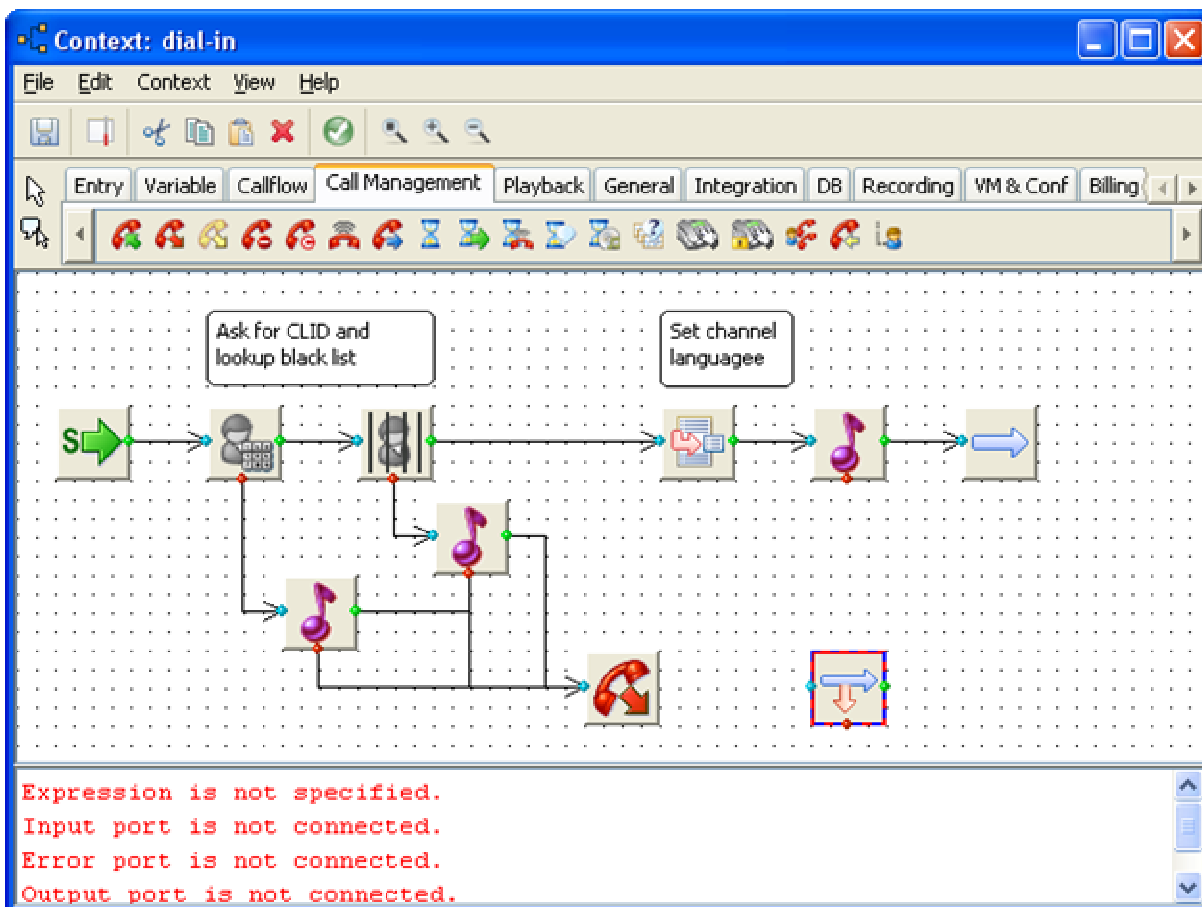
A csatornák azok a kapcsolatok, melyek eljuttatják a hívásokat az Asterisk-ig. Minden hívás egy egyértelműen meghatározott csatornáról jön, vagy csatornára megy ki. Csatorna lehet kapcsolat egy átlagos fejhallgatóhoz vagy telefonvonalhoz, vagy egy logikai híváshoz (mint egy internetes telefonhívás). Minden csatornatípusnak külön konfigurációs fájlja van, így például az *IAX*-nek `iax.conf`, a *SIP*-nek `sip.conf`, az *RTP*-nek pedig `rtp.conf`. Ezekben a fájlokban lehet a szükséges beállításokat eszközölni, valamint ezekben lehet megadni az Asterisk-hez kapcsolódó felhasználókat is.

Külön említést érdemel a `zapata.conf`, melyben Digium és még néhány gyártó kártyáihoz tartozó csatornát lehet beállítani. Ezek a kártyák csatlakoznak a publikus telefonhálózathoz, ugyanis ez nem kivitelezhető egy egyszerű modem segítségével.

A tárcsázási terv konfigurációs fájlja az `extensions.conf`. Ebben a fájlban események vannak összekapcsolva bővítményekkel. Minden bővítményhez egy körülményleírás (context) tartozik, ami lehet az alapértelmezett vagy egy általunk létrehozott, mint például bejövő *SIP* hívások, kimenő távolsági hívások, helyi hívások, irodán belüli hívások vagy valami más. Ezen túl minden kapcsolódó felhasználóhoz is tartozik egy körülményleírás. Ezek beállításai alapján tudja az Asterisk, hogy hogyan kezelje az adott felhasználó által indított hívást, ellenőrizve a felhasználó drága, külső vonalhoz való hozzáférési jogait, valamint más szabályhalmazt alkalmazva a helyi felhasználókra és a külső vonalról érkező kapcsolatokra.

A tárcsázási tervben lehet megadni minden olyan eseményt vagy helyzetet, amit a *PBX*-nek kezelnie kell. Lehet olyan körülményleírást írni, amely csak a nap bizonyos szakában, vagy csak éjszaka működik, illetve lehetőség van külső fájlokból származó kontextus beágyazására is.

Igen bonyolult tárcsázási terveket is létre lehet hozni, ezért ezek készítésének egyszerűsítésére léteznek már grafikus eszközök is.



19. ábra. Egy grafikus tárcsázási terv készítő

Néhány példa, amit a tárcsázási tervvel meg lehet oldani:

- Egy hívást a hangpostára irányítani, ha a hívott felhasználó nem válaszol a megadott időn belül
- Egy hívást több résztvevős konferenciába kapcsolni
- Más Asterisk *PBX*-re irányítani egy hívást
- Az ismeretlen vagy nem kívánt hívóktól érkező hívásokat blokkolni

Van lehetőség a háttérben futó Asterisk-hez való kapcsolódásra, így menedzselve annak működését. Erre létezik egy beépített *CLI*, azaz parancssori interfész, illetve több külső, grafikus interfész is. Ezek segítségével lehetőség van a *PBX*-en történő események, illetve aktív felhasználók, hívások figyelemmel kísérésére; az Asterisk adatbázisának módosítására; a futó *PBX*-be való konfigurációk újratöltésére. Ezen túl egy *TCP* alapú interfész is rendelkezésre áll, melyet az Asterisk-et kiegészítő alkalmazások használhatnak.

6. Összefoglalás

A korábbi fejezetek összefoglalták a *DOCSIS* technológián alapuló hálózat felépítésének követelményeit, lehetőségeit, ismertették összetevőit, az üzemeltetés során hasznos kiegészítő programjait.

Szó volt a fizikai hálózat fizikai és műszaki igényeiről, az eszközök hálózathoz való hozzáféréseinek módjáról, a modemek bekapcsolása után lezajló bejelentkezési folyamatról, az *IP* szintű általános beállításokról, majd a szolgáltatáshoz nem szervesen kapcsolódó, ám az üzemeltetés szempontjából igen fontos és hasznos kiegészítő protokollokról, programokról. Ezt követően a hangszolgáltatás kapcsán beszéltem a PacketCable hálózatról, a *SIP* protokoll, valamint egy ingyenes szoftverről, amely bárki számára lehetővé teszi a *VoIP* megvalósítását.

Az itt leírt ismereteken túl rengeteg további, főként gyakorlati ismeret szükséges egy kábeltelevíziós hálózaton keresztül megvalósított internet- és hangszolgáltatás üzembe helyezéséhez, ugyanakkor úgy gondolom, hogy ez a szakdolgozat jó alapként szolgálhat a hálózat és a hozzá kapcsolódó eszközök, protokollok, szolgáltatások, programok megismeréséhez.

A célom nem egy komplett rendszer össze- és beállításának, parancsról parancsra történő ismertetése volt, hiszen ez igen sok tényezőtől függ, hanem az ehhez szükséges általános ismeretek, protokollok, alkalmazások összegyűjtése. Egy adott rendszer megépítéséhez szükséges beállítások elsősorban a hálózat felépítésétől, a használt eszközöktől, a ráfordítás mértékétől, valamint a hálózattal szemben támasztott elvárásoktól, igényektől függenek.

7. Szójegyzék

- *AGI*: Az Asterisk Gateway Interface segítségével kommunikál az Asterisk *PBX* a felhasználó által létrehozott szkriptekkel. Abban megegyezik a *CGI*-vel, hogy bármely nyelv használható, hiszen a szabványos kimeneten és bemeneten keresztül kommunikál.
- *ASN. 1*: Az Abstract Syntax Number One egy rugalmas, szabványos jelölésrendszer, amely gépfüggő kódolási eljárásoktól független objektumok struktúráinak megadásának módját írja le, formális szabályok gyűjteményeként.
- *Broadband*: A szélessávú internet kapcsolatok esetében alkalmazott jelző.
- *CATV*: Community Antenna TeleVision, azaz közösségi antennás televízió, amely kifejezés elterjedtebb változata a kábeltelevízió szó.
- *CDM*: A kódosztásos multiplexelés lehetővé teszi több jel- vagy bitfolyam számára ugyanazon adatátviteli közeg használatát.
- *CDMA*: Kódosztásos közeghozzáférés esetén két vagy több eszköz egy időben használhatja ugyanazon kommunikációs csatornát. Részletesebb leírásért lásd az 4.2. fejezetet.
- *CDR*: Hívás részletező rekord, amely telefonhívások részletes információit tartalmazza, mint például a hívó és a hívott szám, a híváskezdeményezés kezdete, a hívás időtartama, stb.
- *CGI*: Common Gateway Interface.
- *Channel-bonding*: Egy hoszt két vagy több hálózati interfészének az üzembiztosság vagy az átviteli sebesség növelése érdekében való összekapcsolása esetén használt kifejezés.
- *CLI*: A parancssoros interfész angol nyelvű mozaikszava.
- *CM*: A kábelmodem szó angol nyelvű rövidítése. A kábelmodem egy internet szolgáltatás előfizetői oldalán elhelyezkedő, mely az adatkommunikáció lebonyolításáért felel. Részletes leírása a 3. fejezetben olvasható.
- *CMTS*: A Cable Modem Termination System egy nagysebességű adatátviteli szolgáltatás nyújtására használt központi eszköz, amely a szolgáltató fejállomásán helyezkedik el. Részletes ismertetésért lásd a 3. fejezetet.

- *CPE*: Az előfizetői végpontokon levő modemekre csatlakoztatott eszközök gyűjtőfogalma.
- *DHCP*: A Dynamic Host Control Protocol teszi lehetővé egy IP címtartomány dinamikus kiosztását. A kliens az IP címet csak egy bizonyos időre kapják bérbe, ezért arra *DHCP lease*-ként is szokás hivatkozni. A bérleti időt a *lease time* jelöli. További információ: RFC2131.
- *DHCP lease*: Lásd *DHCP*.
- *DHCP relay*: Az az eljárás, amely során a *DHCP* kérést tartalmazó csomagot a *DHCP relay* ügynök a tényleges *DHCP*-t megvalósító szerverhez továbbítja.
- *Dial-up*: Betárcsázós kapcsolatokra, illetve modemekre használt angol kifejezés.
- *DOCSIS*: Data Over Cable Service Interface Specification, amely a kábeltelevíziós hálózatokon keresztül megvalósított számítógépes hálózat alsó két rétegét leíró specifikáció.
- *Downstream*: A hálózati kommunikáció előre (letöltési) irányára utaló angol kifejezés.
- *Firmware*: A hardvereszközökbe ágyazott és ezekhez szorosan kötődő programok angol elnevezése.
- *FTP*: Fájlátvitel céljából létrehozott protokoll. A részletes ismertetése az RFC959-ben olvasható.
- *H.323*: A *H.323* egy az audió-vizuális kommunikációs *session*-öket leíró rendszer.
- *HFC*: Az angol Hybrid Fibre-Coaxial kifejezés mozaikszava. Az optikai és koaxiális kábelek együttes alkalmazásakor áll elő az ilyen típusú hálózat.
- *IAX*: Az Inter-Asterisk eXchange mozaikszava és az Asterisk nevű *PBX* saját protokollja, melyet több más *PBX* is átvett. További információ az 5.3. fejezetben olvasható.
- *IP*: Az internet egyik alapvető protokollja, amely a TCP/IP referenciamodell hálózati rétegében helyezkedik el. Jelenleg két verziója létezik: az IPv4 és az IPv6, melyek részletes leírása az RFC791-ben és az RFC1883-ban található.
- *IPDT*: Az *Internet Protocol Digital Terminal* kifejezés mozaik szava.

- *IPTV*: Egy olyan rendszer, ahol a digitális televíziós adás az *IP* protokollt használva egy számítógépes hálózati infrastruktúrán keresztül jut el az előfizetőkhöz.
- *ISUP*: Az *ISDN User Part* az *SS7* jeladó protokollgyűjtemény része, melyet a *PSTN* telefonhívásainak felépítésére használnak.
- *MAP*: A MultiAccess Packet-eket a *CMTS* folyamatosan sugározza az előreirányú csatornán. Ezek segítségével tudják a modemek a felcsatlakozásukhoz szükséges paramétereiket beállítani.
- *MER*: A *MER*, azaz a modulációs hibaarány egy olyan mérték, amely egy digitális modulációt igénybe vevő kapcsolat teljesítményének, minőségének mérésére szolgál. A *MER* az egyes szimbólumok eltérését mutatja az ideális helyzetüktől. Minél nagyobb ez az eltérés, annál valószínűbb, hogy a jelet vevő eszköz hibásan detektálja az adott digitális szimbólumot.
A *MER* érték kiszámítására használható képlet: $MER(dB) = 10 \cdot \log_{10}(P_{jel} / P_{hiba})$, ahol *P* a teljesítmény négyzetes átlagát jelöli.
- *MIB*: A Management Information Base kifejezés mozaikszava, amely az *SNMP*-n keresztül menedzselhető eszközök információs adatbázisát jelöli.
- *Minislot*: A *CDMA* közeghozzáférés legkisebb továbbítható egysége. További információ a 4.2. fejezetben olvasható.
- *MTA*: Az analóg telefonok kábelhálózathoz való csatlakoztatását lehetővé tevő eszközt nevezik *MTA*-nak, azaz Multimedia Terminal Adapter-nek. Ez lehet különálló egység és kábelmodembe épített is. Ez utóbbit *E-MTA*-nak is szokás nevezni.
- *NAT*: Olyan hálózati forgalomtovábbító eljárás, amely a *NAT*-ot alkalmazó router átírja a forrás és/vagy cél IP címét, valamint esetleg a portját is a csomagnak. Erre általában azért van szükség, hogy a privát hálózatban levő eszközök csomagjai is kikerülhessen az internetre és vissza. További információért lásd az RFC1631-et.
- *NTP*: A számítógépek óráinak szinkronizálására kifejlesztett protokoll, amely részletes leírását az RFC1305 tartalmazza.
- *NTSC*: Elsősorban Észak-Amerikában használt 6 MHz-es csatorna-sávszélességet használó televíziós szabvány.
- *PAL*: A világ legelterjedtebb televíziós szabványa, 8 MHz-es csatorna-sávszélességgel. Jelenleg Magyarországon is ez a szabvány van használatban.

- *PBX*: A *Private Branch eXchange* feladata a privát hálózaton belüli telefonok egymással való összekötése, valamint összeköttetés biztosítása számukra a *PSTN*-el.
- *PDU*: A *Protocol Data Unit* az adott protokoll által értelmezhető fejlécből, és az általa becsomagolt adatból áll.
- *PNG*: Képek tárolására és tömörítésére alkalmas fájlformátum.
- *PSTN*: A világ publikus, vonalkapcsolt telefonhálózatára használt, angol nyelvű mozaikszó.
- *QAM*: Quadrature amplitude modulation. Bővebb információért lásd az 4.1. fejezetet.
- *QoS*: A Quality of Service egy erőforrás foglalatást vezérlő mechanizmus, mellyel különböző alkalmazásokhoz, felhasználókhöz, illetve adatfolyamokhoz különböző prioritást rendelhető. További információért lásd az RFC2990 dokumentumot.
- *QPSK*: Quadrature Phase-Shift Keying. Bővebb információ az 4.1. fejezetben olvasható.
- *RADIUS*: Hálózati autentikációs protokoll. Lásd: RFC2866.
- *Ranging*: Az a folyamat, mely során a kábelmodem felméri a *CMTS*-től való távolságát, és ennek megfelelően beállítja az adási jelszintjét.
- *RFI*: Radio Frequency Interface, amely a *CMTS*-ek, illetve a kábelmodemek koaxiális interfészére utal.
- *RTP*: Valós idejű hang és videó továbbító protokoll, amely egy szabványos csomagformátumot határoz meg ezek továbbítására. Leírását lásd az RFC1889-ben.
- *RSVP*: Az erőforrás fogláló protokoll, ahogy az nevéből is sejthető, *QoS* igények jelzésére használatos. Definiálást az RFC2205 tartalmazza.
- *S-CDMA*: Synchronous Code Division Multiple Access. További információért lásd az 5.2. fejezetet.
- *SDP*: A Session Description Protocol az internetes multimédiás közvetítés inicializálási paramétereit leíró formátum, mely az RFC4566-ban definiált.
- *SECAM*: Főként Franciaországban használt televíziós szabvány, mely 8 MHz-es csatorna-sávszélességet használ.

- *Session:* A *session* egy fél-permanens interaktív információcsere két vagy több kommunikáló eszköz között.
- *SGMP:* Az RFC1028-ben definiált Simple Gateway Monitoring Protocol célját tekintve az *SNMP* elődjének tekinthető.
- *SID:* A Session ID segítségével tudják az eszközök az összefüggő adatkommunikációt megvalósítani.
- *SIP:* Session Initialization Protocol. Részletes leírását lásd az 5.2. fejezetben és az RFC3261-ben.
- *SNMP:* A Simple Network Management Protocol a hálózati eszközök (switchek, *CMTS*-ek, kábelmodemek, stb.) menedzselésére használatos protokoll. Részletes ismertetése megtalálható a 4.4.3.1-es fejezetben, illetve az RFC3411-ben.
- *SNR:* A jel és a háttérzaj teljesítményének hányadosa, melyet egy kommunikációs csatorna teljesítményének, minőségének mérésére használnak. A kiszámítására használatos képlet a következő: $SNR(\text{dB}) = 10 \cdot \log_{10}(P_{\text{jel}} / P_{\text{zaj}}) = 20 \cdot \log_{10}(A_{\text{jel}} / A_{\text{zaj}})$, ahol P az átlagos teljesítmény, A az amplitúdók négyzetes átlagát jelöli.
- *Splitter:* A *splitter* egy egyszerű eszköz egy koaxiális kábel több ágra való bontására.
- *SS7:* A *Signaling System 7* több telefonos jeladó protokoll gyűjteménye, melyet a világ publikus, vonalkapcsolt telefonhálózatainak nagy része használ.
- *TCP:* A hálózati kommunikáció egyik legfőbb protokollja, mely megbízható, sorrendhelyes adatfolyam továbbítást tesz lehetővé. További információ az RFC793-ban olvasható.
- *TDM:* Az időosztásos multiplexelés angol nyelvű mozaikszava. Ezen multiplexelési mód esetén két vagy több jel- vagy bitfolyam egy kommunikációs csatornán történő szimultán továbbítására használatos. A csatornára egyszerre csak egy folyam jelei/bitjei kerülnek rá.
- *TDMA:* Az időosztásos közeghozzáférés lehetővé teszi több eszköz számára ugyanazon kommunikációs csatorna használatát. Részletesebb információért lásd az 5.2. fejezetet.
- *TFTP:* Trivial File Transfer Protocol. Részletesebb leírásért lásd a 4.4.2. fejezetet, valamint az RFC1350-et.

- *Time slot:* Az időrés angol megnevezése, amely a *TDM*-hez és a *TDMA*-hoz kapcsolódik. Ezek az eljárások az adatátviteli közeget bizonyos időre „bérbe adják” az azt használni kívánó folyamatoknak, eszközöknek. Azt az időt, amíg ugyanazon folyamat/eszköz birtokolja a csatornát, időrésnek nevezzük.
- *ToD:* Time of Day. Lásd 4.4.1. fejezet.
- *UCD:* Az Upstream Channel Descriptor, azaz a visszirányú csatornaleíró tartalmazza a visszirány paramétereit, köztük a használandó modulációt, csatorna sáv szélességet és a frekvenciát.
- *UDP:* A hálózati kommunikáció egyik alapprotokollja, amely segítségével a csomagok eljuttathatók a megfelelő alkalmazáshoz. Az *UDP* fejléc csak alapvető információkat tartalmaz, mely így nem biztosít sorrendhelyes kézbesítést. További ismertető: RFC768.
- *Upstream:* A hálózati kommunikáció vissz (feltöltési) irányára utaló angol kifejezés.
- *USSNR:* Upstream Signal To Noise Ratio. A visszirányú csatorna jel/zaj arányának rövidítése. További információért lásd az *SNR*-t.
- *VoIP:* A Voice over Internet Protocol egy az interneten vagy más csomagkapcsolt hálózatokon való hangátvitelre tervezett protokoll.

8. Irodalomjegyzék

Könyvek:

- [1] **Barth, Wolfgang:** Nagios: System and Network Monitoring. San Francisco: No Starch Press, 2006.
- [2] **Meggelen, Jim Van; Madsen, Leif; Smith, Jared:** Asterisk: The Future of Telephony. O'Reilly, 2007.
- [3] **Stallings, William:** SNMP, SNMPv2, SNMPv3, and RMON 1 and 2. Boston, Mass. [etc.]: Addison-Wesley, 2000.
- [4] **Tanenbaum, Andrew S.:** Számítógép-hálózatok. Budapest: Panem, 2004.

Hálózati forrás:

- [5] **Agilent Technologies:** Digital Modulation in Communications Systems – An Introduction. In <http://cp.literature.agilent.com/litweb/pdf/5965-7160E.pdf>
Felhasználás időpontja: 2008-02-18
- [6] **Buda, Fabien; Lemois, Emmanuel; Sari, Hikmet:** An Analysis of the TDMA and S-CDMA Technologies of DOCSIS 2.0. 2002. In http://cn.juniper.net/solutions/literature/white_papers/200032.pdf
Felhasználás időpontja: 2008-03-04
- [7] **CableLabs:** Data-Over-Cable Service Interface Specifications, DOCSIS 2.0, Radio Frequency Interface Specification, CM-SP-RFIv2.0-I12-071206. In <http://www.cablelabs.com/specifications/archives/CM-SP-RFI2.0-I12-071206.pdf>
Felhasználás időpontja: 2008-02-23
- [8] **CableLabs:** Data-Over-Cable Service Interface Specifications, DOCSIS 3.0, MAC and Upper Layer Protocols Interface Specification, CM-SP-MULPIv3.0-I06-071206. In <http://www.cablelabs.com/specifications/archives/CM-SP-MULPIv3.0-I06-071206.pdf>
Felhasználás időpontja: 2008-02-23
- [9] **CableLabs:** PacketCable 1.5 Architecture Framework Technical Report, PKT-TR-ARCH1.5-V01-050128. In <http://www.cablelabs.com/specifications/archives/PKT-TR-ARCH1.5-V01-050128.pdf>
Felhasználás időpontja: 2008-04-12

[10] **Gróf Róbert:** Adatátviteli szolgáltatások kábeltelevízió hálózaton. 2005. In <http://www.kabelkon.hu/download/kabelmodem%20rendszer.pdf>

Felhasználás időpontja: 2008-02-07

[11] **Ötvös Tamás; Solti Miklós:** A kábeltelevíziós átviteltechnika alapjai. 2008. In http://jegyzet.sth.sze.hu/ftp!/Tais_cuccok/Interaktiv.kabeltv/IRODALOM_I/KTVJEGYZ/CATV/CATV0003.DOC

Felhasználás időpontja: 2008-03-16

RFC dokumentumok (<http://www.rfc-editor.org>):

[12] **Case, J.; Fedor, M.; Schoffstall, M.; Davin, J.:** A Simple Network Management Protocol (SNMP). 1990. In RFC1157.

[13] **Rosenberg, J; Schulzrinne, H.; Camarillo, G.:** SIP: Session Initiation Protocol. 2002. In RFC3261.

[14] **Sollins, K.:** The TFTP Protocol (Revision 2). 1992. In RFC783.