**1. Overview**

**Hogyan növelheti egy szolgáltató hálózatmenedzsmenttel a bevételeit?**

Három különböző esetben növelheti a bevételeket a hálózatmenedzsment. Az első esetben maga a **kiadások csökkentése** nyit meg egy teljesen új piacot. Ez akkor lehetséges, ha a kezdő költségek a menedzsment nélkül túl nagyok lennének ahhoz, hogy az a felhasználónak megérje, azonban alacsonyabb költséggel új szolgáltatásokat lehet nyújtani. A második eset, amikor a menedzsment a szolgáltatás megrendelése és üzembe helyezése közti **időtartamot rövidíti le**, ezzel gyorsabban generál bevételeket. Bizonyos esetekben a gyors szolgáltatás hiánya az ügyfél teljes elveszítésével járhat. A harmadik esetben a szolgáltató **menedzsment funkciókkal bővíti a szolgáltatásait**, melyek így több ügyfelet vonzanak.

**Milyen szereplők lehetnek a hálózatmenedzsment területen? Ismertesse ezek közül a hálózatmenedzsment felhasználóit!**

Az összes felhasználó csoportra igaz: fontos a sebesség, a QoS, a hibakezelés, a kihasználtság. A felhasználók első csoportja a **távközlési vagy hálózati szolgáltatók**. Ezeknek a bevételi forrása az általuk nyújtott szolgáltatás, és a hálózatmenedzsment létfontosságú számukra, mivel a költségeik túlnyomó része a működési költség. Ezen túl a bevételeiket is növelhetik, ha gyorsan tudnak szolgáltatást üzembe helyezni (provisioning), esetleg új szolgáltatásokat nyújtani a menedzsment által csökkentett induló költségekkel (truck roll), vagy akár menedzsment szolgáltatásokkal bővített szolgáltatást nyújtani.

A második csoport a **céges IT részleg**. A fő különbségek a szolgáltatókhoz képest: az IT a cég számára kizárólag költséget jelent, nem céljuk a bevétel generálása; csak egy ügyfelük van, és az ügyfelüknek nincs igazán lehetőségük a leváltásukra (csak az outsourcing); mivel a cég számára nem alaptevékenység az IT, nem fontos a versenyképességük, ezért sokszor valóban outsourcingolják a szolgáltatásaik nagy részét; alacsony (vagy néhol semmilyen) törvényi szabályozás vonatkozik rájuk. A harmadik csoport a **végfelhasználók**, akik itt a hálózatmenedzserek. Ide tartoznak a hálózati rendszergazdák, technikusok, help-deskesek, és a hálózattervezők is. Szerepük és működésük erősen a cég jellegétől függ.

**Hasonlítsa össze üzleti hálózatmenedzsment lehetőségek szempontjából a berendezésgyártót és a független hálózatmenedzsment alkalmazások készítőjét!**

A berendezés gyártói általában minimális képességű menedzsment alkalmazásokat szeretnének nyújtani, hiszen nekik az elsődleges céljuk a „vas” eladása. Azonban az utóbbi évek trendje az, hogy a különböző gyártók eszközei között egyre inkább csak a nyújtott szolgáltatás minősége a különbség, és ezért egyre nagyobb figyelmet szentelnek a hálózatmenedzsment lehetőségek bővítésére. A független hálózatmenedzsment alkalmazások készítői alapvetően azt az űrt próbálják betölteni, amely az eszközök gyártói és a hálózatüzemeltetők között van. Ez azt jelenti, hogy a gyártó által nyújtott minimális menedzsment felület helyett, egy könnyebben kezelhető, több funkcionalitást is tartalmazó interfészt nyújtanak. További előnyük, hogy a független (harmadik fél) alkalmazások gyártó függetlenek és több gyártó több eszközének is hasonló interfészt nyújt az üzemeltetők felé.

**Milyen módon járulhat hozzá az üzleti sikerhez a hálózatmenedzsment? Hogyan?**

Cost, Quality, Revenue

● magas rendelkezésre állás

● redundáns hardware, kommunikációs utak

● végpont-végpont szolgáltatás

● monitorozás

● korreláció analízis a riasztásokra.

● ha nem tudják garantálni a szolgáltatás minőségét, ellenszolgáltatás

**Mit csinál a "hálózatmenedzsment"?**

A hálózatmenedzsment azokat a cselekvéseket, eljárásokat, folyamatokat és eszközöket jelenti, amelyek egy hálózatban annak működtetésével (operation), adminisztrációjával (administration), karbantartásával (maintenance), és szolgáltatásával (provisioning) kapcsolatos. **OAMP**

**Hálózatmenedzsment feladatok milyen alkalmazási működéssel valósíthatóak meg? Hogyan és milyen hálózatmenedzsment célokat szolgálnak?**

**Tranzakció-alapú** (transaction based) működés során a hálózat kéréseket küld ki, válaszokat dolgoz fel, valamint folyamatokat menedzsel. Alapvetően a szolgáltatás felépítésénél (Provisioning) használt módszer. **Megszakítás-alapú** (interrupt) működés kvázi valós idejű karakterisztikával a monitoring során használt. Riasztások (alarmok) segítségével tudja a hálózat pontos állapotát megmutatni a menedzsernek, és megfelelő esetekben figyelmeztetni a hibákra. **Adatmennyiség-alapú** (number cruncing) működés a statisztikai elemzések szempontjából fontos, hogy megfelelően tudjuk elosztani az erőforrásokat (bottleneck), hogy meghatározzuk az Accounting szempontjából a kapcsolódó bejegyzéseket, hogy tervezhessünk a jövőbeni bővítésekről.

**Miért tekinthetjük a hálózatmenedzsmentet egy elosztott rendszernek! Érveljen**

A hálózatmenedzsment alkalmazások definíció szerint elosztottak, ugyanis vannak menedzselt és menedzselő rendszerek. Emellett a skálázhatóság, megbízhatóság és magas rendelkezésre állás eléréséért maga a menedzselő rendszer is elosztott kell, hogy legyen. Például a karbantartás csak akkor valósítható meg egy rendszeren, ha a feladatait át tudják venni tőle. Elosztott rendszerre a nap járását követő menedzsment-szolgáltatások is jó példa, ugyanis a munkaidőben mindig érdemes a helyileg közel lévő rendszereket használni.

**Mi az a "swivel-chair syndrome" (görgős szék szindróma)? Jó ez vagy rossz? Miért?**

A görgős szék probléma arra utal, amikor egy rendszergazda görgős székével közlekedik a több terminál között. A probléma lényege az integráció hiánya, melynek következtében a felhasználók kénytelenek rendszeresen sok együtt nem működő alkalmazás között váltogatni. Ez rossz, és elkerülendő, mivel a felhasználóknak rendkívül sok különböző alkalmazás használatát kell megtanítani, ami költséges és felesleges. Helyette az integrált menedzsment lenne a megoldás, melynek lényege, hogy közös megjelenítő felületen, közös adatbázisokban dolgozna a több különböző menedzsment-alkalmazás.

**Milyen szereplők lehetnek a hálózatmenedzsment területen? Ismertesse ezek közül a hálózatmenedzsment előállítóit!**

A berendezés gyártói általában minimális képességű menedzsment alkalmazásokat szeretnének nyújtani, hiszen nekik az elsődleges céljuk a „vas” eladása. Azonban az utóbbi évek trendje az, hogy a különböző gyártók eszközei között egyre inkább csak a nyújtott szolgáltatás minősége a különbség, és ezért egyre nagyobb figyelmet szentelnek a hálózatmenedzsment lehetőségek bővítésére. A független hálózatmenedzsment alkalmazások készítői alapvetően azt az űrt próbálják betölteni, amely az eszközök gyártói és a hálózatüzemeltetők között van. Ez azt jelenti, hogy a gyártó által nyújtott alkalmazások általában nem alkalmasak több gyártó berendezéseinek menedzselésére, vagy ha igen, akkor is csak korlátozottan. A független (harmadik fél) alkalmazások azonban megpróbálnak több gyártó több eszközének közös interfészt nyújtani az üzemeltetők felé. A rendszer integrátorok a különböző eszközök és alkalmazások egyéni testre szabását végzik, mivel általában az alkalmazások nem működnek gond nélkül együtt, és ezért külön menedzsment szolgáltatást tudnak nyújtani.

**Hogyan segítheti a hálózatmenedzsment a vállalati megtakarításokat?**

* Hálózat tesztelő és hibaelhárító eszközökkel
* Szolgáltatások indítását megkönnyítő rendszerekkel
* Teljesítmény monitorozó eszközökkel és szűk keresztmetszet (bottleneck) analízissel

A megtakarításokat a működési költségek csökkentésével tudja a hálózatmenedzsment növelni. Ez jelenthet hálózati tesztelő és hibaelhárító eszközöket, melyek megtakarítást jelentenek a rendszergazdák idejéből, automatikus üzembe helyezést, mely az egyébként sok egymást követő redundáns információkat igénylő lépést helyettesíti, vagy teljesítmény-analizátorokat, melyek segítenek a pontosabb erőforrás-elosztásban.

**Mit takar az OAMP rövidítés? Röviden ismertesse az egyes betükhöz rendelt feladatokat!**

Az OAMP az Operation, Administration, Maintenance és Provisioning kifejezéseket takarja, melyek a hálózatmenedzsment alap funkciói.

**O**: a hálózat (és az általa nyújtott szolgáltatások) fennakadás nélküli működtetése. Beletartozik a hálózat folyamatos ellenőrzése (monitoring), hogy a hibákat a lehető leghamarabb észrevegyék, lehetőleg mielőtt egy felhasználónak problémája származna belőle.

**A**: az erőforrások és azok kiosztásának számon tartása. Minden „házimunka” ide tartozik, mely ahhoz kell, hogy irányításunk alatt tartsuk a rendszert.

**M**: javítások és fejlesztések. Ide tartoznak a módosító és megelőző intézkedések is, a menedzselt hálózat jobbá tétele érdekében.

**P**: az erőforrások oly átrendezése, hogy egy adott szolgáltatást lehetővé tegyenek

**2. Basics**

**Oldja fel a MIB akronimát. Fejtse ki mi az a MIB!**

Management Information Base: A MIB egy virtuális adattár, mely menedzsment nézetet tartalmaz az adott eszközhöz. Nem egy valódi adatbázis, hanem csak egy nézet, bejegyzései mögött valódi konfigurációs lehetőségek vannak.

**Oldja fel az NOC akronimát. Mi történik az NOC-ban?**

Network Operations Center: Az a hely, ahonnan egy nagy hálózatot menedzselnek, minden menedzsment funkciót innen indítanak. Kiterjedt hálózatokban nem elég egy NOC, hanem akár több, regionális NOC-ra is szükség van (akár a nap járása szerint). A menedzsmenttel kapcsolatos tevékenységeket itt végzik.

* monitorozás
* provisioning
* hálózati eszközök konfigurációjának biztonsági mentése
* felhasználók adatainak gyűjtése számlázás céljából
* kommunikációs berendezéseket is tarthatnak itt

**Hasonlítsa össze a manager/agent és a client/server paradigmát hasonlóságok és különbségek alapján!**

Ebben a kontextusban a manager a client és az agent a server, olyan értelemben, hogy a manager kér információkat az agent-től, és az kiszolgálja azt. A különbség az, hogy míg alapértelmezésben a server kevés, client sok a felállás, itt a manager a kevés és az agent sok.

**Miért nem kezelhető a MIB egy adatbázisként?**

A MIB számára az adatbázis-kezelők túl robusztus, túl költséges megoldást jelentenének, nagy overheaddel az eszközök számára. Mivel a MIB-ek információi általában hierarchikus elrendezésűek, valamint elosztottak, nehezen tudná egy adatbázis megragadni ezeket a különlegességeket. Fontos még, hogy míg egy adatbázisban sok hasonló struktúrájú adat van kevés táblában, addig egy MIB-ben sok fajta bejegyzésből van kevés. Lényegében a MIB nem lehet valódi adatbázis, hiszen folyamatosan változik a természeténél fogva.

**Oldja fel az OSS akronimát. Mit csinál egy OSS?**

Operational Support Systems. Az OSS maga a menedzsment rendszer, általában a szolgáltatók nevezik így azért, mert lényegében a működésüket támogatja a menedzsment rendszer.

**Ha egy hálózatnak 99.999% (5 kilences) rendelkezésre állást kell biztosítania, akkor szükséges-e a menedzsment rendszernek is 99.999%-os rendelkezésre állást biztosítani? Indokolja válaszát!**

A hálózat működéséhez nem szükséges a management. Az 5 kilences rendelkezésre állást nem szabad, hogy befolyásolja az, ha a management rendszer rendelkezésre állása alacsonyabb. Ezért ahhoz, hogy a hálózat 5 kilences rendelkezésre állást biztosítson, általában nem kell, hogy a management rendszer is ezt tudja biztosítani. Azonban szükséges megkülönböztetni egymástól a management alkalmazások típusait: az olyan management rendszerek rendelkezésre állása, amely a hálózat monitorozásával foglalkozik, nem szabad, hogy sokkal az 5 kilences rendelkezésre állás alá essen, mert a monitorozás kihagyása jelentős hatással lehet a hálózat rendelkezésre állására. Másrészről, például egy olyan management rendszer rendelkezésre állása, amely azzal foglalkozik, hogy új felhasználókat adjon a hálózathoz, kevésbé kritikus. Végül, az olyan management rendszer rendelkezésre állása, amely számlázási adatokat gyűjt, ugyanolyan magas rendelkezésre állást kell, hogy biztosítson, mint a hálózat. Nem azért, hogy a hálózat rendelkezésre állása nagyobb legyen, hanem azért, mert a számlázási információk elvesztése jelentős bevételi veszteség a szolgáltató számára.

**Az ágens kifejezése milyen két kontextusban használatos? Hogyan kapcsolódnak ezek egymáshoz?**

Az ágens kifejezés jelentése az egyik kontextusban a menedzser-ágens kapcsolatban a menedzselt eszközök szerepe. A másik jelentése egy szoftver, amelyet a hálózati elemek nyújtanak, és megvalósítja a menedzsment interfészt.

**Milyen hálózatot lehet használni a hálózatmenedzsmentre? Mik az egyes megközelítések előnyei és hátrányai?**

Lehet használni a hasznos forgalmat lebonyolító hálózattal közös, **megosztott hálózatot**, vagy **dedikált menedzsment hálózatot**. Előbbi olcsóbb megoldás, és sokszor az egyetlen, mivel nincs lehetőség egy másik önálló hálózat kiépítésére. Utóbbi előnyei a megbízhatóság, a biztonság, az interferencia elkerülése és a hálózattervezés egyszerűsítése.

**3. Dimensions**

**Milyen rétegekre (TMN) bonthatjuk fel a hálózatmenedzsmentet? Mi az egyes réteget feladata? Milyen előnyök és hátrányokkal járhat ez a rétegződés?**

* **Network Element:** A hálózati berendezések által nyújtott menedzsment funkciók.
* **Element Management:** A hálózati berendezések állapotának megfigyelése, változtatása és azok öntesztelésre utasítása illetve az általuk küldött riasztások figyelése.
* **Network Management:** A hálózati berendezések közti kapcsolatok és függőségek megfigyelése és a hálózat működésének szempontjából helyes beállítása.
* **Service Management:** A hálózat által nyújtott szolgáltatások menedzsmentje, és annak biztosítása, hogy a szolgáltatások megfelelően működnek és jól futnak.
* **Business Management:** A szolgáltatásokkal kapcsolatos üzlet menedzsmentje úgy, mint számlázás, helpdesk menedzsment, üzleti előrejelzések, stb.

Előnye a modellnek, hogy egymásra épülő funkciókra lehet bontani a hálózatmenedzsmentet. Hátránya, hogy ha szó szerint próbáljuk megvalósítani, sokszor feleslegesen sok adminisztrációt igénylő és rugalmatlan rendszert kapunk.

**Adja meg a menedzsment életciklust és röviden ismertesse egyes fázisat!**

Az egyes modulok kialakulásának, a hálózat élettörténetének lekövetése.

Fázisai:

(Prepare) -> Plan -> (Design) -> Deployment -> Operate -> (Optimization) -> Decommission

**Előadáson nem voltak a zárójelesek!**

* **Prepare:** az elérni kívánt cél meghatározása, technológiai döntések meghatározása, előnyök és hátrányok elemzése történik ebben a fázisban.
* **Plan:** megvalósítással kapcsolatos kérdések megvitatása (pl. pontosan mit kell újítani, van-e hely az új eszköznek, bírni fogja-e az áramellátás az új eszköz beüzemelését)
* **Deployment:** Ez azt jelenti, hogy a berendezéseket el kell helyezni, majd beüzemelni. Egyedi management műveletek végrehajtását is magában foglalja (pl. IP cím beállítása)
* **Operation:** Az a fázis, amelyben a legtipikusabb hálózat management feladatok foglalnak helyet (pl. hálózat monitorozása, hibakeresés, teljesítmény növeléséért tett lépések, teljesítménnyel kapcsolatos statisztikai adatok gyűjtése, számlázási adatok gyűjtése stb.)
* **Decommission:** Különféle okok miatt szükség lehet az eszközök leszerelésére (pl. újabb eszközökre cseréljük a régieket). Az eszközök leszerelését úgy kell végrehajtani, hogy az esetleges hálózati kimaradás a lehető legkisebb fennakadást okozza.

**Milyen dimenziói vannak a hálózatmenedzsmentnek? Sorolja fel ezeket!**

* **Interoperability** (együttműködés) – kommunikáció, funkciószolgáltatás, információszolgáltatás, közös dimenziók
* **Subject** (tárgy) – scope, technológia, szolgáltatások
* **Life cycle** (életciklus) – plan, deploy, operate, decommisson
* **Layers** (rétegek) – minden rétegekre van osztva, management layer, network element -> element manager -> network management -> service manager -> business manager
* **Functions** (funkciók) – layerenként különböző, ahhoz hasonló FCAPS funkciók (Fault, config, accoutnt, performance, security)
* **Process & Organization** (folyamat és szervezés)

**Hálózatmenedzsment együttműködés szempontjából milyen szempontokat kell figyelembe venni?**

* **Kommunikáció:** A kommunikáló felek egymás üzeneteit megértik-e, milyen szabályok szerint építjük fel az üzeneteinket? (fő részei: session establishment, authentication, request type, timestamp, encoding)
* **Funkció:** Milyen funkciókat várunk el az agenttől? Pl. passzív működés (monitoring), aktív működés (QoS beállítás); lehetséges-e az “event subscription”; meg tudja-e mondani, hogy milyen feladatokra képes stb.
* **Információ:** Definiálja azt a modellt, hogy hogyan akarunk kommunikálni.

**Miért kellenek "sablonok" (cookbooks), dokumentáció és standard eljárások a hálózatmenedzsmentben?**

A sablonok, a dokumentációk, a standard eljárások azért szükségesek, hogy a szervezés és minden management eljárás (vagyis maga a network management) a lehető leghatékonyabb legyen.

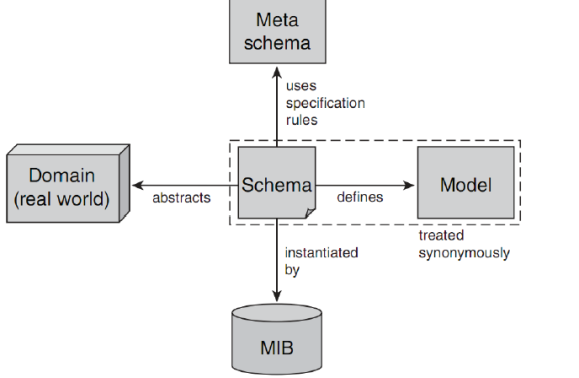
**4. Management Information**

**Mi a lényeges különbség az objektum típus és az objetum példány között? Érveljen egy példán keresztül!**

An object type is a definition of a kind of managed object; it is declarative in nature. In contrast, an object instance is an instantiation of an object type which has been bound to a value. For example, the notion of an entry in a routing table might be defined in the MIB. Such a notion corresponds to an object type; individual entries in a particular routing table which exist at some time are object instances of that object type.

Tehát az objektum típus egy menedzselt objektumot definiál, míg az objektum példány egy objektum típus példányosítása konkrét értékekkel. MIB = instance

**Hogyan lehet MIB-eket definiálni? Hogyan viszonyul egymáshoz a való világ, a schema, a modell, a meta schema, és a MIB? Rajzoljon is!**

* Domain: A domaint a modell foglalja össze (absztrahálja). A való világot domainnek hívjuk, mert ez jelképezi azt a területet, amelyet modellezünk.
* Meta Schema: specifikációs nyelv, ennek szabályait kell használni MIB definíció során. Meghatározza, hogy hogyan kell írni és értelmezni definíciókat.
* Schema: A modell, ami alapját képezi a MIB-ben lévő management információnak egy MIB definícióban.
* MIB: A séma, amely az eszköz MIB-jében van példányosítva.
* Model: A séma meghatározza a modellt, a sémával hasonló módon kezelik.

**Mi az a SMI? Mit ad meg?**

Structure of Management Information (SMI): A MIB definíciókat MIB modulokként specifikálja.

An MIB module generally serves a particular purpose, such as to define management information related to a device’s communication interfaces or to a voice-mail server feature that is embedded on a particular type of device.

* Internet:
  + MIB-2: RFC 1213
  + SMI: RFC 1155
  + SMIv2: RFC 2578
* MIB definitions in MIB modules
  + Particular purpose
  + MIB of a devices instantiates multiple MIB modules
* SNMP MIB consists of a set of managed objects that instantiate object types that are part of a MIB module.

**Együttműködés a mgmnt rendszerek között -> ennek érdekében a mgmt adatokat strukturáltan érdemes tárolni.**

**Milyen objektum típusok lehetnek az Internetes MIB-ekben?**

* **Skalár:** Csak egyszer példányosított objektumok
* **Oszlopos:** Többször példányosított objektumok

Nincsenek komplex adattípusok, minden objektum egyszerű típusként van definiálva az SMI-ben és az SMIv2-ben. Ezekkel kell minden információt reprezentálni. Az objektumok mindig a menedzsment fa levélelemei.

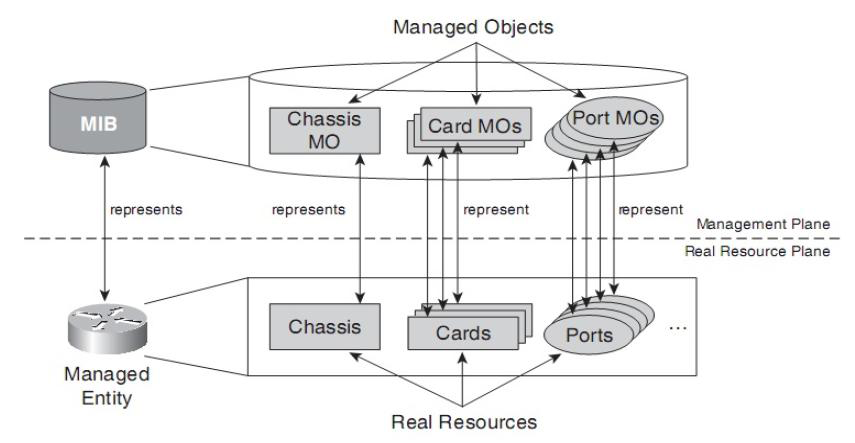
A gyártók hozzáadhatják a saját eszközeikre vonatkozó MIB modulokat külön engedélyek nélkül.

**Objektum orientált modellezést használnak-e az SNMP MIB-ben? Indokolja a válaszát!**

**Nem.**

A MIB objektumok nem rendelkeznek néhány olyan képességgel, amelyet általában az objektum orientáltsággal társítunk. Ilyen képesség az öröklés, a polimorfizmus (egy alosztály példányainak az a tulajdonsága, mi szerint tekinthetünk rájuk úgy, mint az ősosztály példányaira) és metódusok osztály definíciójába történő beágyazása. Tehát az SNMP MIB objektumok egyszerű MIB változók.

**Hogyan viszonyul egymáshoz a MIB, MO, Managed Entity, Real Resources? Rajzoljon!**



**Milyen kategóriái lehetnek a menedzsment információknak?**

* Állapot információk
  + fizikai és logikai erőforrások
  + működési adatok
  + gyorsan változhatnak az adatok
  + management alkalmazás nem tudja módosítani az állapotinformációt, csak lekérni
* Fizikai konfigurációs adatok
  + az eszközhöz tartozik
  + a manager nem tudja megváltoztatni
  + ritkán változik az információ
  + “Best to cache it at the managers”
* Logikai konfigurációs adatok
  + tipikusan a management által kontrollált és megváltoztatható
  + “Cache it” - dokumentálás, csak akkor változik, ha a manager megváltoztatja
  + Típusai: startup és transient
* Történeti információk
  + nem tükrözi a jelenlegi állapotokat

**Milyen különböző paradigmák szerint lehet a menedzsment modellt megadni? Mutassa be a különbségeket!**

* objektum orientált konstrukció
  + MO osztályok
    - attribútumok
    - figyelmeztetés küldése
  + származtatott osztályok: ősosztály, alosztály, öröklődés
* táblák és változók különféle módon csoportosítva
  + Tábla = M.O. osztály
    - az eszköz egy bizonyos nézőpontja
    - az attribútumok a tábla oszlopai
    - a példányok a tábla sorai
* Mindent parancsként, funkcióként, és ezek paramétereként modellez. Pl. CLI parancsok és ezek felparaméterezése. Itt a modell nincs expliciten megadva.

**Az SNMP MIB rendezésére használt fa miben tér el pl. egy fájlrendszer fától?**

A fájlrendszerben a könyvtár is egy elem tulajdonságokkal (pl.: jogosultságok), MIB fában csak a leveleknek van tulajdonsága, értéke.

**4c. ASN.1 & BER**

**Mi az a BER? Hogyan kapcsolódik az ASN.1-hez? Hogyan működik?**

A BER-t széles körben használják átviteli szintaxisok megadására.

Basic Encoding Rules (BER)

* Kódolási specifikáció
  + CCITT (X.209) és ISO (ISO 8825) szabvány
* **Megadja az octet string kódolását minden ASN.1 típusnak**
* **Típus-hossz-érték type-length-value (TLV) struktúrát használ ( |Type|Length| Value |)**
  + Az ASN.1-es TLV struktúra az rekurzív lehet
    - Minden érték további TLV értékeket tartalmazhat
  + Három metódus a kódolásra:
    - Primitive, definite-length encoding
    - Constructed, definite-length encoding
    - Constructed, indefinite-length encoding
* A használt metódus függ az ASN.1 típustól és hogy a hossza ismert-e a típusból.

**Mi az ASN.1? Mire használják a menedzsmentnél? Hogyan köti össze a helyi adatokat, a megjelenítést, a transzport réteget? Rajzoljon!!!**

SNMP-ben és OSI menedzsmentben használják, az ANS.1 kiterjedten használt alkalmazás adatok és PDU-k definíciójára.

**Abstract Syntax Notation One** (ASN.1)

* Eszköz független adatleíró nyelv
* CCITT (X.208) és ISO (ISO 8824) szabvány
* Absztrakt szintaxist ad meg alkalmazás adatoknak
* Alkalmazási és prezentációs struktúrát ad meg (protocol data units (PDUs))
* Az SNMP és OSI Management Information Base (MIB) megadására



**4.d. Standard MIBs**

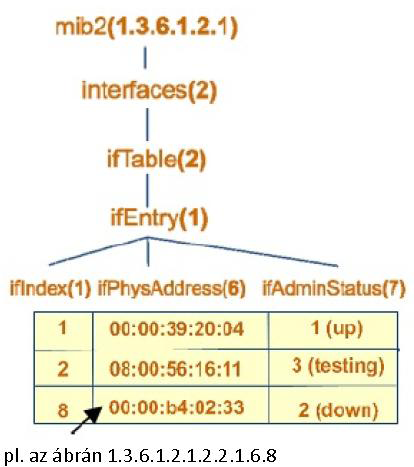
**Válasszon 3 objektumot a MIB-II alatt és röviden ismertesse, hogy milyen menedzsment információkat szolgáltat az alatta lévő fa.**

**system:** a rendszer általános információi (Uptime, Name, Location, stb.)

**interfaces:** információk a rendszer és a hálózat közti interfészekről

**at:** leírja a hálózat és az internet közötti címfordít

**továbbiak:** ip, icmp, tcp, udp, egp, transmission, snmp



**Hogyan épülnek fel a táblázatok a MIB-ben? Milyen ASN.1 típusként vannak definiálva? Mi a MIB konvenció? Hogyan férhetünk hozzá egy meghatározott táblázat elemhez? Rajzoljon és magyarázza!**

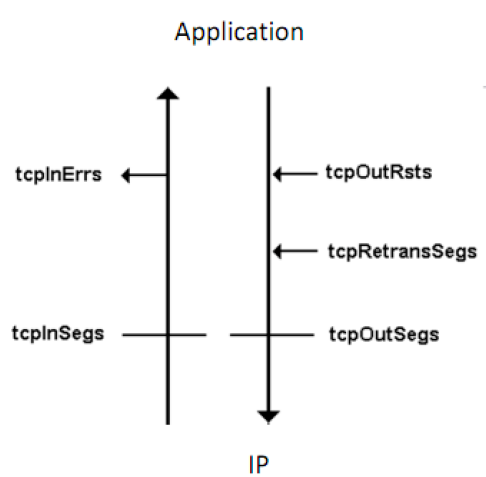
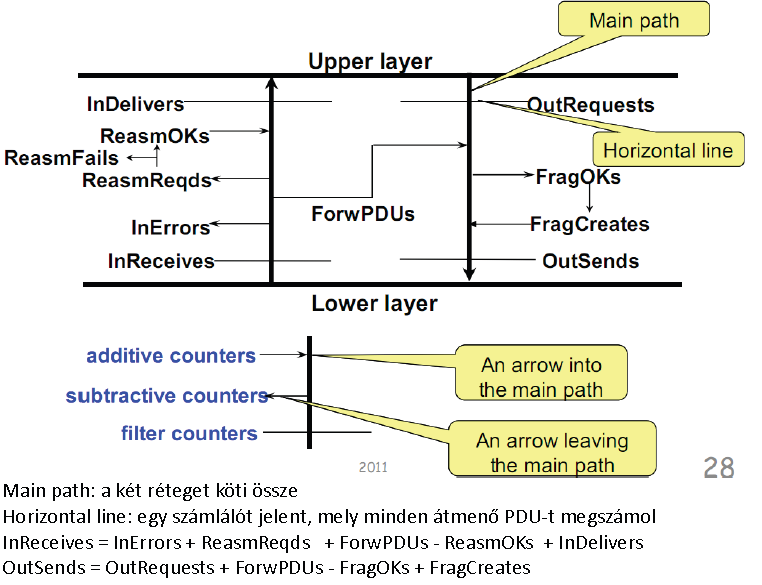
**Kétdimenziós táblázatokat** hozunk létre, amelyek nem tartalmazhatnak beágyazott táblázatokat. Az elemek egyértelmű azonosításáért egy vagy több index a felelős. A táblázat sorai **SEQUENCE típusként** vannak definiálva, a táblázat pedig ezen sorokból alkotott SEQUENCE OF típusból áll össze. Egyszerű objektumok példányaihoz az <objektum\_azonosító>.0 azonosítóval, a táblázat példányaihoz pedig a <táblázat azonosító>.<oszlop>.<indexérték> azonosítóval.

**Ismertesse a CASE diagram koncepciót! Rajzoljon egy mintapédát illusztrálásra, értelmezze a számlálókat!**

A CASE diagram egy jól használható eszköz a MIBek fejlesztésénél

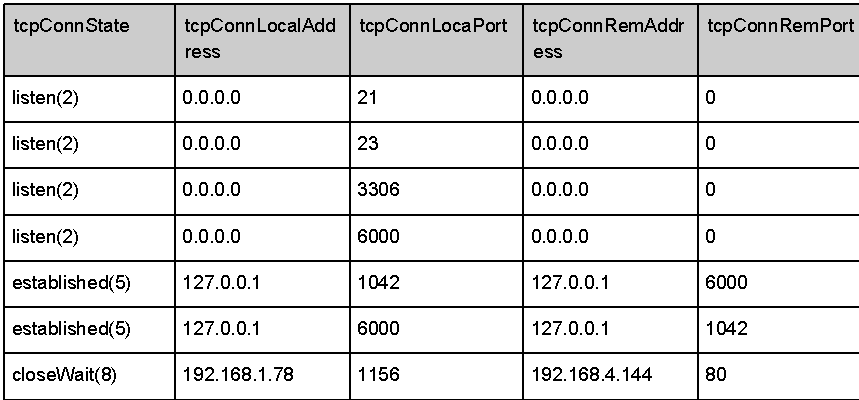
* a csomagtovábbítás folyamának leírása rétegenként
* általában szükséges a forgalmi összetétel számlálása protokoll rétegenként
* úgy, hogy minden valamelyik rétegből küldött vagy fogadott PDU valahová tartozik helyes és helytelen (hibás csomag) egyaránt csomagok áramlása az adott rétegben

ötlet: Jeffrey Case (1989)



**Hogyan lehet több indexet használni egy táblázatban? Hogyan lehet nem egyszerű típusú MO-val indexelni?**

An SNMP table can be defined as an ordered collection of objects consisting of zero or more rows. Each row may contain one or more objects. Each object in a table is identified using the table index. A table can have a single index or multiple indices. A scalar variable has a single instance and is identified by its ".0". On the other hand, a table object or the columnar variable can have one or more instances and is identified by its index value. To identify a specific columnar variable, the index of the row has to be appended to its OID. For example for a table with OID .1.3.6.1.2.1.x.x.xTable, with the column name yy and the index value ind1, the value of the column yy can be got by appending the instance ind1 to the columnar OID .1.3.6.1.2.1.x.x.xTable.xEntry.yy. If the table has multiple indices namely ind1 and ind2 then the value of the column yy can be got by using the OID .1.3.6.1.2.1.x.x.xTable.xEntry.yy.ind1.ind2. For example, consider tcpConnTable. It has four indices namely tcpConnLocalAddress, tcpConnLocalPort, tcpConnRemAddress, and tcpConnRemPort where the values of the table are as follows.



To get the value of the column tcpConnState for the last row, you have to query with the OID tcpConnState.192.168.1.78.1156.192.168.4.144.80 where 192.168.1.78 is the value of tcpConnLocalAddress for the last row, 1156 is the value of tcpConnLocalPort for the last row 192.168.4.144 is the value of tcpConnRemAddress for the last row 80 is the value of tcpConnRemPort for the last row. Also if the index is of integer type, it can be in any order. For example in a table, if the values of the index column are {1,2,3,4}, it can have values in any order say {2,4,3,1}. Lexikografikus rendezéssel?

**Mondjon egy olyan MIB objektumot, ahol write only MAX ACCESS jogosultságnak lenne értelme! Miért?**

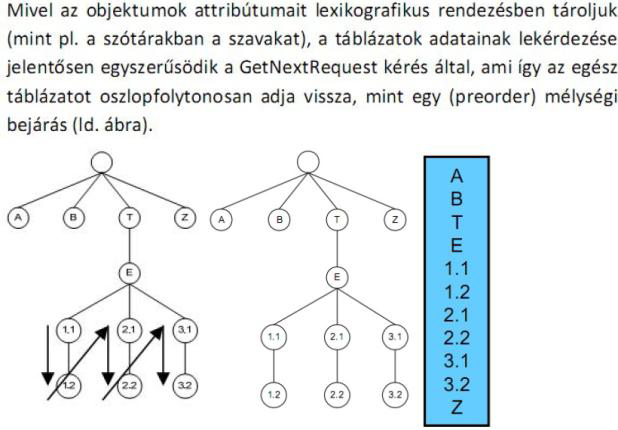
Jelszavaknál van értelme, ugyanis fontos, hogy azokat senki se olvashassa, azonban a jelszó változtatására szükség van.

**A MIB-ben mi a különbség az objektum típus és az instance között?**

Egy objektum típusra vonatkozó OID globálisan egyedi. Egy instance-re vonatkozó OID csak az őt tartalmazó MIB-ben egyedi.

**Ismertesse a lexicographikus sorrendezést! Rajzoljon is! Mire és hol használják?**

Használat: ismeretlen MIB (rész) szekvenciális bejárására (felderítésére).



**Miért van szükség táblázatok indexelésére?**

Mert az indexek azonosítják a táblázat sorait, elemei ezek segítségével érhetőek el. Indexeléssel a táblázat elemei gyorsabban, és nem csak lineáris kereséssel érhetőek el.

**5. SNMPv1**

**Milyen SNMPv1 PDU-k vannak? Ismertesse őket röviden!**

PDU = Protocoll Data Unit

- Get Request : objektum értékének kinyerésére az ágensből

- Get-Next Request: hasonlóan a Get Request, de a MIB fában lexikographikus sorrendben következő objektum értékével tér vissza

- Set Request: objektum értékének beállítására

- Response: ágens válasza a Get Request, Get-Next Request és Set Request PDU–kra

- Trap: ágens jelentésekre, csak egyirányú, nincs válasz a menedzsertől

**Hogyan kérdezne le egy ismeretlen táblázatot SNMPv1-ben?**

Lexikografikus rendezés -> sok getNextRequest

**Milyen biztonsági mechanizmus van az SNMPv1-ben? Hogyan működik? Mi a hibája?**

- Authentication service: Ágens korlátozhatja a hozzáférést a MIB-hez

- Access policy: Különböző menedzsereknek különböző hozzáférési jogosultság

- Proxy service Ágens közvetíthet más menedzselt eszköz felé A menedzselt eszközre vonatkozóan authentication service és access policy kellhet a proxyban

- Az SNMP csak egy primitív és korlátozott biztonsági képességgel rendelkezik

**!community**

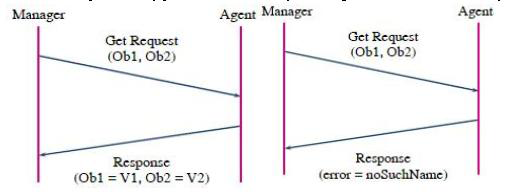
- Ágens és a menedzserek között: authentication, access control & proxy characteristics

- Ágens helyi értelmezése a fentieknek: Minden community egy egyedi community névvel azonosítva, Ágens több community-t is használhat, Minden parancsnak hordoznia kell a community azonosítót, Különböző ágensek használhatják ugyanazt a community azonosítót

- SNMP authentication service, Minden SNMP üzenetnek hordoznia kell a community nevet (mint valami jelszó), Nagyon primitív megközelítés, Ezért a legtöbb ágens csak GET parancsot fogad el, text alapú biztonsági mechanizmus gyenge biztonság, community, ágens kérheti minden üzenetre ,de ez egyszerű textként utazik az üzenetben.

**Mit jelent az SNMPv1 üzenetek atomikus volta?**

Vagy minden értéket visszaad, vagy egy sem kerül visszaadásra. Sikeres (bal kép); ha bármelyik objektum nincs implementálva, sikertelen (jobb kép):



**Mi az, hogyan és hol használják a lexikografikus sorrendezést?**

- Soros MIB hozzáféréshez

- A fa struktúrából következően, bármely objektum OID-je meghatározható a gyökértől az objektumig vezető úton

- lexicographical ordering: preorder traversal (root, left, right) of a tree depth-first search (esetleg korábbi példa)

**6. RMON**

**Milyen informácitó szolgáltat az RMON statistic csoport?**

MAC (alhálózat) szintű kihasználtság és hiba statisztika., hálózat terheléséről,

R/W objektumok: etherStatsSource, etherStatsOwner, etherStatsStatus

counterek: packets, octets, broadcasts, multicasts, collisions, errors, csomagméret eloszlások

**Mi az általános kapcsolat az adat és a vezérlési táblák között?**

A vezérlési tábla indexei (pl. rm1ControlIndex) segítségével az adattáblában azonosíthatjuk az azonos bejegyzéshez tartozó adatsorokat, mivel a saját indexükön kívül tartalmazzák a hozzájuk tartozó kontrolltábla-bejegyzés indexét is (pl. rm1DataControlIndex).

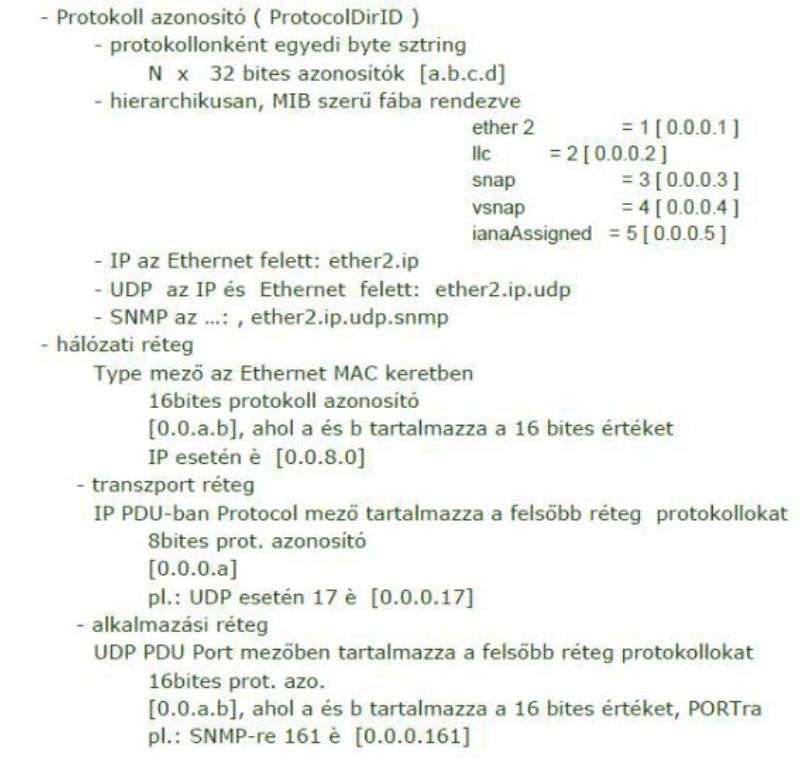
**Milyen informácitó szolgáltat az RMON history csoport?**

Periodikus statisztikus minták a statistics csoportra. (kivétel a csomagvesztés-eloszlás) Körkörös puffert (bucket) használ, melynek méretét a menedzser igényli. Vezérlés: historyControlTable (melyik szegmensre, milyen mintavétellel – javaslat: 30mp és 30p) Adatok: etherHistoryTable

**Milyen informácitó szolgáltnak az RMON host csoportok?**

Hoszt-forgalmak az adott alhálózaton. Ki-/bemenő csomagok, oktettek, kimenő multicast, broadcast üzenetek, hibák. A táblázat indexelhető MAC cím (hostTable), létrehozási idő (hostTimetable) vagy valamely paraméter (hostTopN) alapján.

**Mely rétegekben működhet az RMON-2? Hogyan adhatjuk meg, hogy mely protokollokat vizsgáljuk?**



**Milyen riasztási sémákat alkalmazhatunk RMON-nál?**

- monitor vagy menedzser új sor létrehozásával hozhat létre új riasztást

- figyelt változó, mintavételi intervallum, küszöb soronként egyedi

- emelkedő (rising) küszöbátlépés ha az aktuális minta nagyobb egyenlő mint a felső küszöb és az utolsó érték kisebb a küszöbértéknél

- eső (falling) küszöbátlépés ha az aktuális minta kisebb egyenlő mint az alsó küszöb és az utolsó érték nagyobb a küszöbértéknél

- kétféle érték a riasztásokhoz

- absoluteValue : mintavételi időpontokban

- deltaValue: rate of change

**Hogyan kezeli az RMON probe a konkurens vezérlő tábla sor hozzáadást? Mi az RMON polka?**

RMON polka:

1. menedzser sor létrehozása createRequest(2) EntryStatus mezőértékkel

2. ha az ágens végrehajtotta a műveletet, akkor a sor státuszt underCreation(3) értékre állítja (mindaddig ez az értéke, amíg a menedzser az összes sorát be nem állítja)

3. ha a menedzser kész az összes sorával, azok státuszát valid(1) értékre állítja

4. ha a sor már létezik, vagy createRequest végrehajtása alatt van, akkor hibával tér vissza

**Mi az RMON koncepció? Milyen előnyökkel jár(hat) a távoli monitorozás?**

**A dedikált (vagy egyéb funkciókat is ellátó) RMON eszközök** (monitorok vagy probe-ok) a menedzser és az ágensek tehermentesítése érdekében promiscuous módban **figyelik az adott alhálózat forgalmát, csomag(részleteket) gyűjtve, analizálva** (proaktív monitorozás). Így nem csak az egyedi eszközökre, hanem az adott LAN-ra vonatkozóan is gyűjthetünk információkat. Csökkenti az SNMP forgalmat, több menedzserrel is együttműködhet (növekszik a megbízhatóság), az aktív analízisek alapján gyorsabb diagnosztika és jelentés az NMS felé, valamint lehetőség van offline monitorozásra is (ha a menedzser épp nem elérhető).

**Hogyan működik az RMON szűrési csoportja?**

Remote Network Monitoring

- csomagok kiválasztására

- Kétféle szűrő (filter)

- Adat szűrő: bitmintán alapuló illesztés

-Státusz szűrő: csomagstátusz alapján(crchiba, jó csomag, stb)

- szűrő logika: input = bejövő csomag rész, szűrésre

filterPktData = tesztelendő bitminta

filterPktDataMask = bitmaszk, releváns bitek

filterPktDataNotMask = teszt egyezés vagy nem egyezésre

Példa:

filterPktDatOffset = 0

filterPktData = 0x0000000000A50000000000BB

filterPktDataMask = 0xFFFFFFFFFFFFFFFFFFFFFFFF

filterPktDataNotMask = 0x000000000000FFFFFFFFFFFF

**Hogyan vezérelhetjük a távoli monitorozást? Mik ennek jellegzetes problémái? Hogyan kezelik ezt az SNMP keretben?**

Cél a távoli monitorozás, adatgyűjtés, alhálózati viselkedés megfigyelése, rendszer és ágens tehermentesítés. probléma a felismereés, és a jelentés, együttműködés többb menedzserrel.

Az **RMON MIB**-ben kerültek meghatározásra **a vezérlési funkciók** (Configuration Control, Action Invocation), ezek funkcionális csoportokba vannak rendezve, csoportonként kontrol- és adattáblákkal (utóbbiak csak olvashatók). A kontrol táblában határozzuk meg az adatgyűjtés paramétereit (forrás, adat típusa, gyűjtés időtartama), az adattáblákból pedig kiolvashatjuk az összegyűjtött adatokat.

A kontrol táblában néhány paramétert csak úgy módosíthatunk, hogy invaliddá tesszük (ezzel

töröljük a hozzá tartozó adattábla bejegyzéseket is), majd a módosított paraméterekkel újra

létrehozunk egy bejegyzést.

**Hogyan valósítja meg az RMON probe az erőforrások hatékony megosztását? Mire kell figyelnie a menedzsereknek?**

Minden monitorozási funkcióhoz tulajdonost rendel (owner, aki létrehozta), így a menedzser felismerheti saját foglalásait (ha nincs már rá szüksége, felszabadíthatja). Egyeztetés és megfelelő jogok birtokában más erőforrását is felszabadíthatjuk (pl. ha a foglalást birtokló menedzser időközben összeomlott). A monitor rendelkezik saját funkciókkal is, ezekben a legnagyobb a bizalma. Egy funkció kérés beérkeztekor a kontroll tábla végigpásztázása hasonló, már kért funkció után.

**Milyen információt szolgáltnak az RMON matrix csoport?**

Kihasználtság, és hiba statisztika host párokra.

Hoszt párokra tárol forgalmi információkat az alhálózatra vonatkozóan : packets, octects, errors

* mátrix formájú tárolás
* 1 kontroll tábla
* 2 adattábla
  + ugyan az az információ
  + de az egyik forrás és cél szerint, a másik cél és forrás szerint indexelve
* az összes hosztból kifelé irányú és az összes a hosztba befelé irányú forgalom kigyűjthető

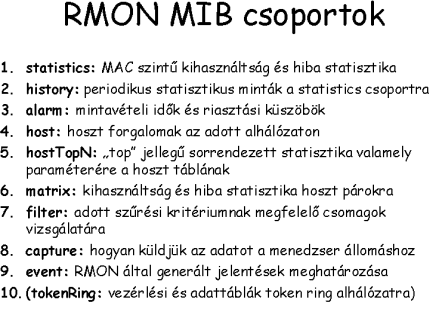
**Hogyan működik a csomagelkapás az RMON-ban?**

Elfogó (capture) csoport segítségével. A csatornán (szűrőkön) átjutó csomagokat puffereli két táblázattal:

\_bufferControlTable: pufferelési funkciók

\_captureBufferTable: adatok tárolására

Egy ControlTable bejegyzéshez tartozik n db. BufferTable bejegyzés.



**Hogyan épül egymásra az RMON adat és statisztika gyűjtés? Milyen kapcsolódó csoportok vannak?**

Az RMON az adatokból vett minta alapján statisztikákat generál, majd ezen statisztikák periodikus mintavétele alapján top- és táblázatos historykat készít.

**7. RMON-2**

**Hogyan lehet egy alkalmazási réteg protokoll statisztikáit gyűjteni RMON2-ben?**

RMON2-ben 2 csoport foglalkozik hosztonkénti statisztika gyűjtéssel:

○ hálózati réteg hoszt csoport (nlHost)

○ alkalmazási réteg hoszt csoport (alHost)

Mindkét csoport adattábláját az nlHost kontroll táblája (nlHostControlTable) vezérli.

alHost: Minden egyes hálózati címen, minden egyes felismert alkalmazási réteg protokollra egy bejegyzés keletkezik az nlHostTable-ben, ha a protocolDirAlHostConfig==supportedOn. Mindkét irányú forgalmat számolja, de csak helyes MAC keretekre. Pl. adott hoszton az MS Mail forgalom

**Mi a különbség az RMON-1 és RMON-2 között?**

- RMON MIB kiterjesztése a MAC réteg fölé: OSI rétegek 3 – 7

- hálózati réteg protokollok és hálózati címek szerinti monitorozás

- alkalmazás szintű monitorozás: email, file transzfer, WWW protokollok

RMON1:

- MAC címek szerint / LAN szegmens

- a forgalom forrása nem meghatározható pl. routeren érkező forgalom esetén

RMON2:

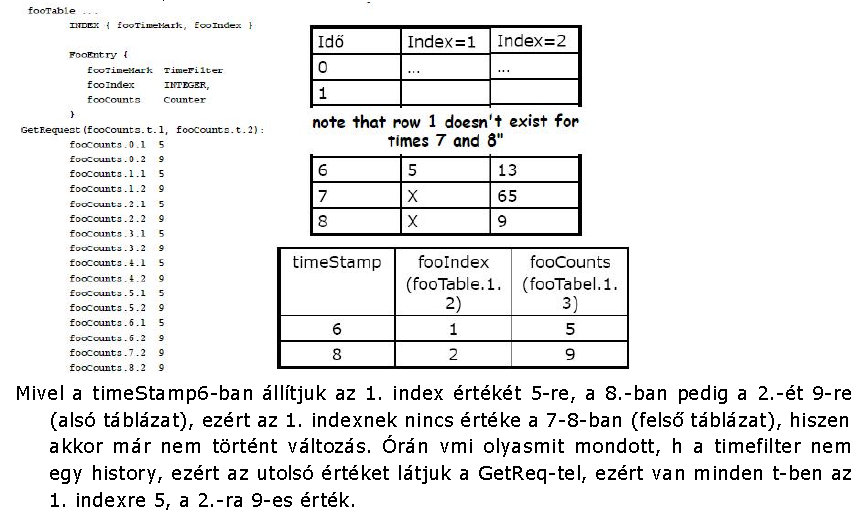
- MAC keretben a magasabb szintű fejrész értelmezése, tipikusan IP csomag

- megállapítható az „igazi” forgalom forrása / célja

**Hogyan működik a TimeFilter az RMON2-ben?**

Time filter indexing: időre történő szűrés

Mivel periodikus lekérdezések esetén a változások érdekesek, csak azokat az objektumokra történik szűrés, amelyeknél változás történt az utolsó lekérdezéshez viszonyítva.



**Hogyan adhatunk meg protokollokat RMON2-ben?**

Az RMON2 MIB **protokoll könyvtárcsoportja** (protocolDir) valósítja meg a protokollokra vonatkozó típusinformációk központi tárolását, így a menedzser megismerheti, hogy mely protokollok értelmezésére képes az ágens. Az RMON2 protocol directory group egy protokoll könyvtár tábla, amely minden kezelendő protokollra egy bejegyzést tartalmaz.

**8. SNMPv2**

**Mi az az Augments? Mire használják? Hogyan?**

Oszlopokat lehet meglévő táblákhoz adni és ezzel bővíteni.

**Sor hozzáadás, követelmények:**

- PDU-nál nagyobb sor létrehozása

- Ágensben nem implementált oszlopok megismerése a menedzsernek

- Az ágensben a menedzser számára el nem érhető oszlopok felderítése

- Menedzserek konkurens sor kezelése

- Sor létrehozásának védelme (sorrendiség)

- tooBig észlelése a végrehajtás előtt

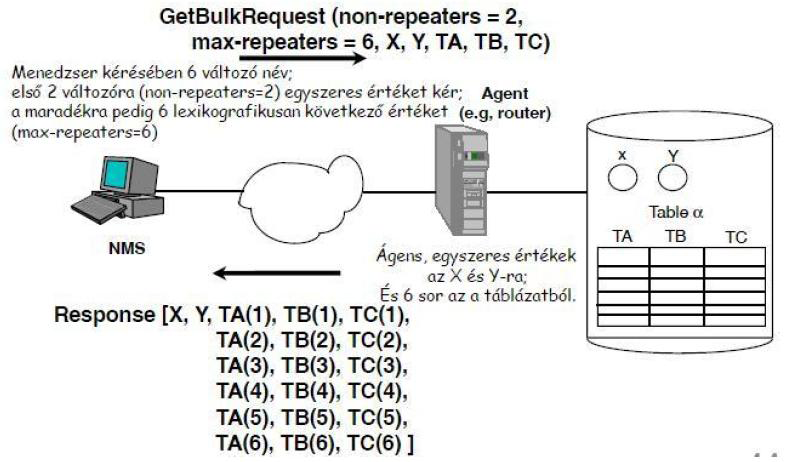
- Egyazon sorban read-only és read-create objektumok létezése

- Ezek mind: fontos vagy hasznos tulajdonságok

**Mire használják az InformRequest PDU-t?**

Az InformRequest v2 menedzsertől másik v2 menedzserhez **küldött kérés**, a menedzsment alkalmazás nevében valamilyen információ kinyerésére.

Célja: hierarchikus és elosztott menedzsment támogatása, ahol egynél több menedzsert használnak.

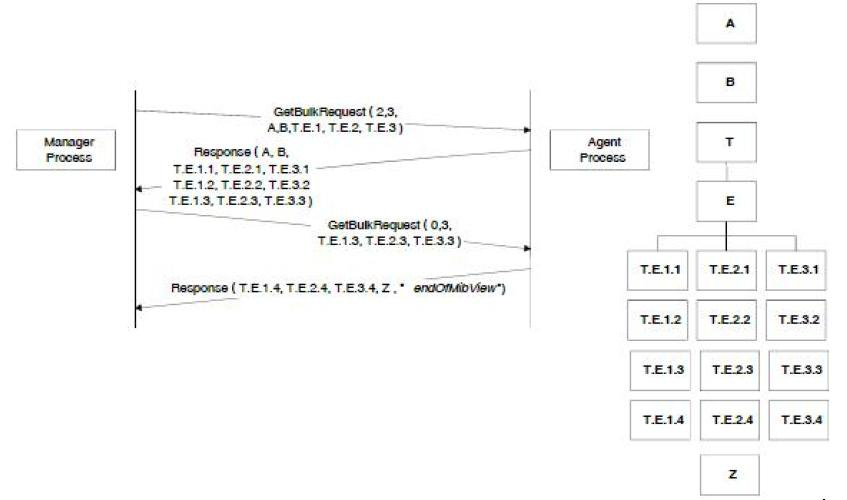
**Ismertesse a GetBulkRequest üzenet működését! Rajzoljon példát is!**

A kérés-válaszok minimalizálásra ha „nagy” mennyiségű adatot kell lekérdezni

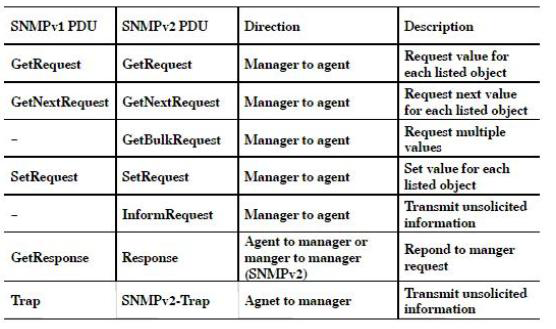
- Az OID hozzárendelés hasonló a GetNextRequest-hez, Lexikografikus következő elemek

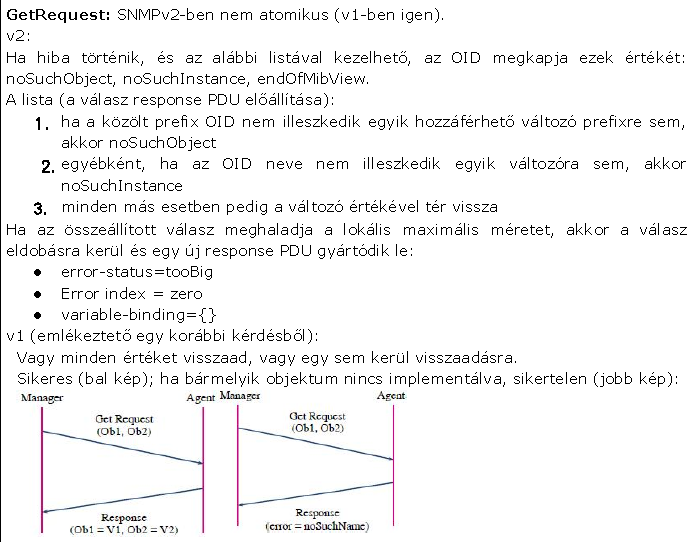
- N + R változó név, N egyedi érték lekérdezésre, R többszörös érték lekérdezésére

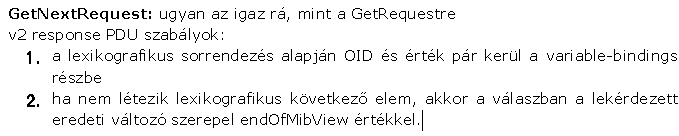
- non-repeaters és max-repetition, Az N és R jelzésére

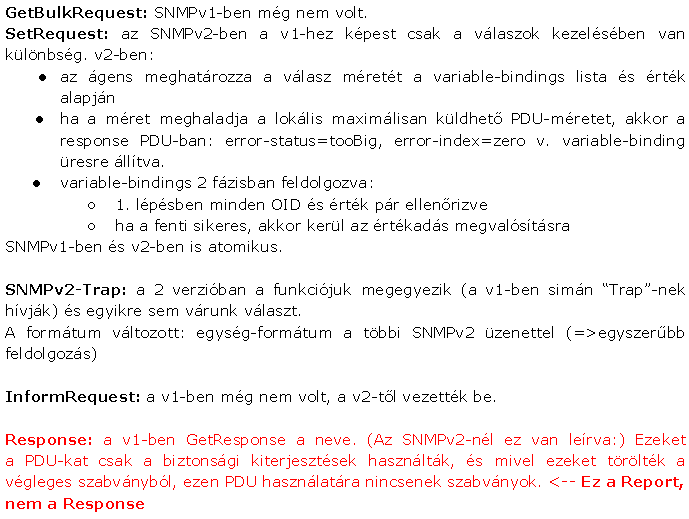


**Hasonlítsa össze az SNMPv1 és v2 üzenet típusokat! Miben különböznek ezek? Miért?**









**Hogyan lehet új sort létrehozni egy táblázatban SNMPv2-ben?**

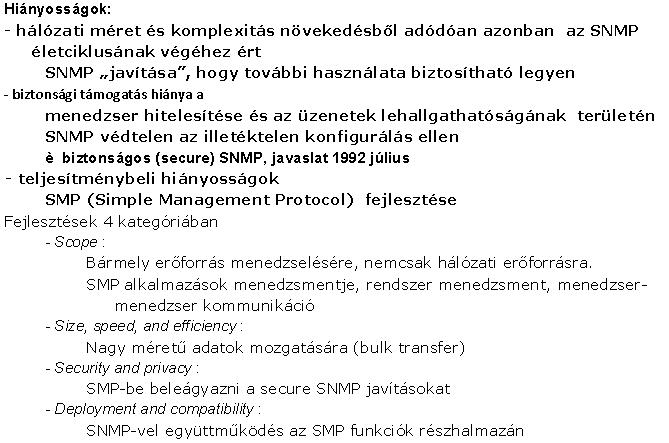
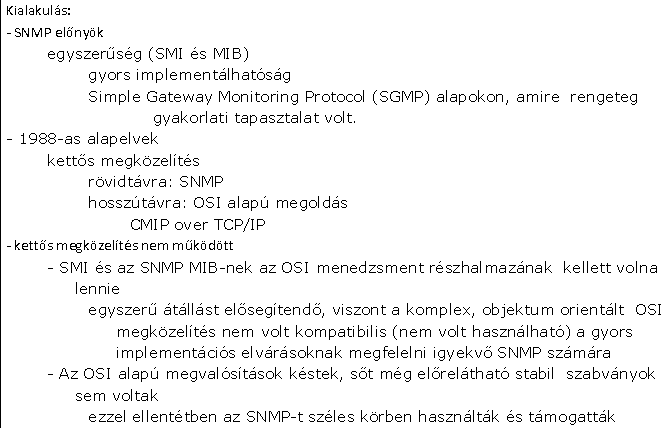
RMON-nal.

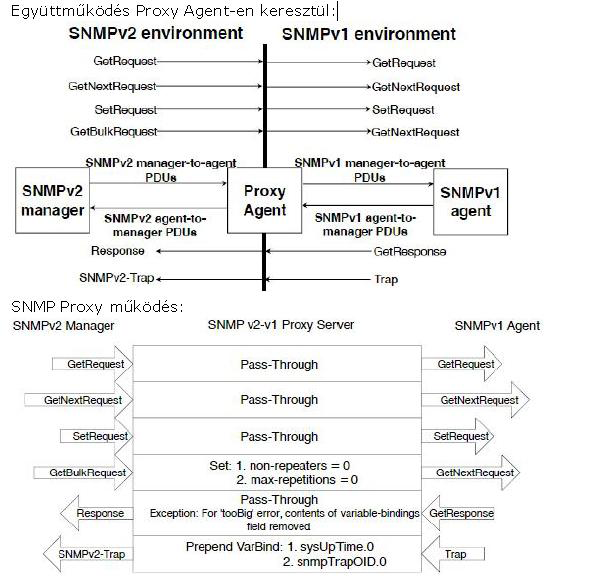
2 fajta tábla van az SNMPv2-ben:

● menedzser által sor hozzáadható és törölhető

● menedzser által nem vezérelhető táblák (teljesen az ágens által vezérelve)

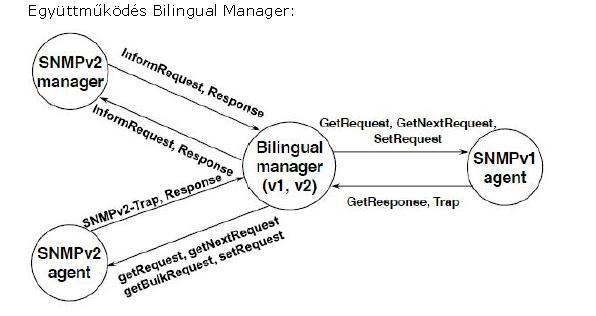
**Hogyan alakult ki az SNMPv2-es szabvány csoport? Milyen v1 hiányosságokat javít?**



**Hogyan működhet együtt egy SNMPv1 és v2-es rendszer? Milyen üzeneteket kell és hogyan átalakítani?**

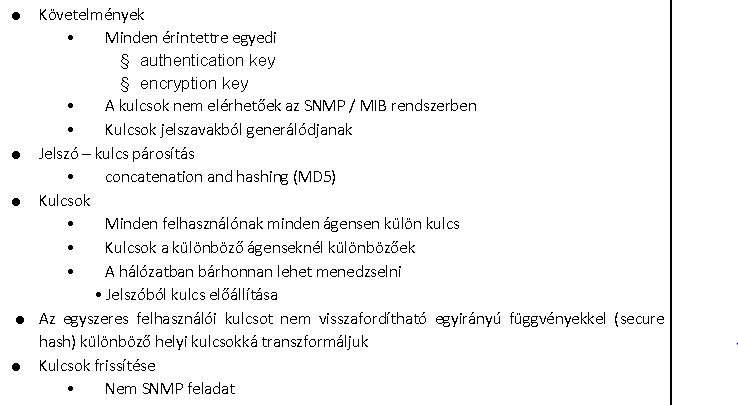
Visszamenőleges kompatibilitás szükségszerű; v2 menedzserek v1 ágensekkel.

* Két kategóriát kell tekinteni
* Menedzsment információkat
* Protokoll működést
* SMI különbségek
  + Object definitions
  + Trap definitions
  + Compliance definitions
  + Capabilities definitions



**9. SNMPv3**

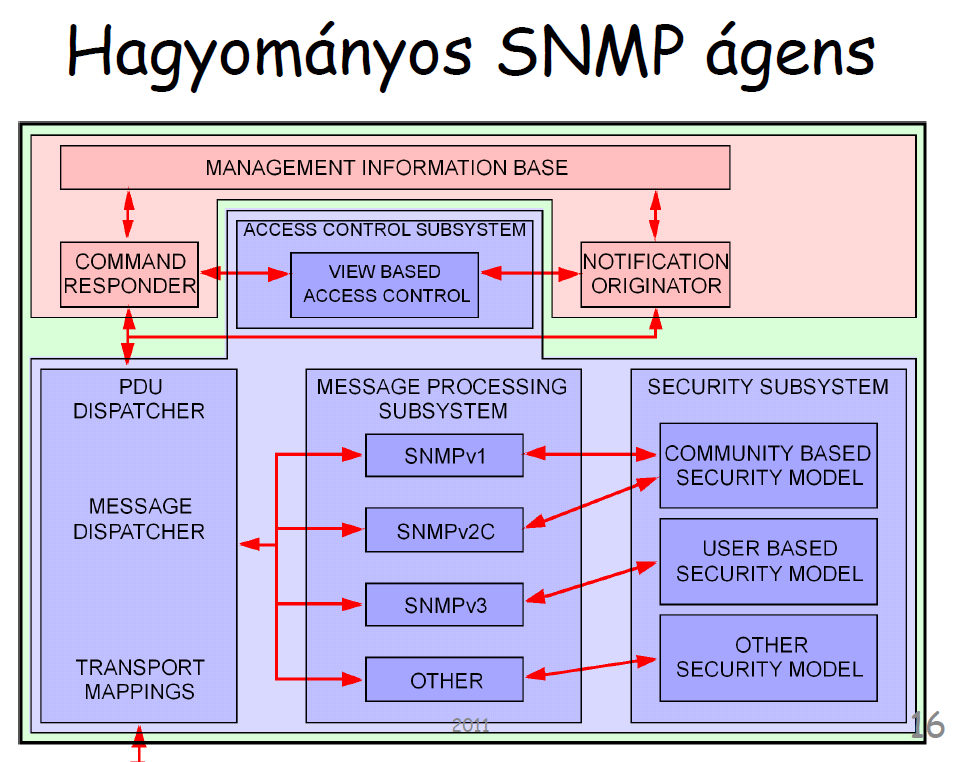
**Hogyan kezelik a felhasználói kulcsokat SNMPv3 architektúrában?**



**Milyen biztonsági módjai vannak az SNMPv3-nak?**

NoAuthNoPriv, AuthNoPriv, AuthPriv

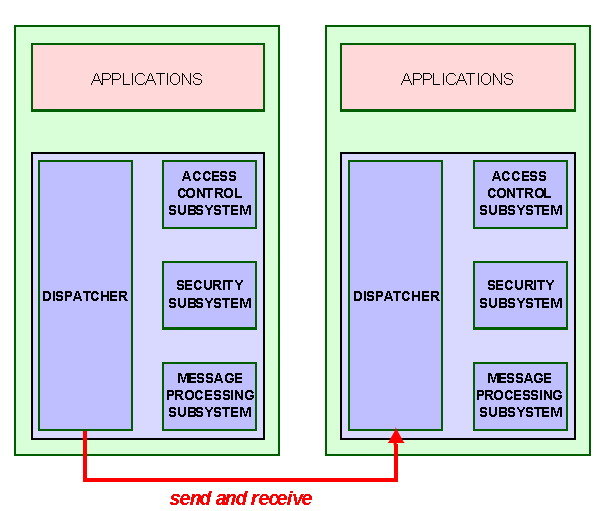
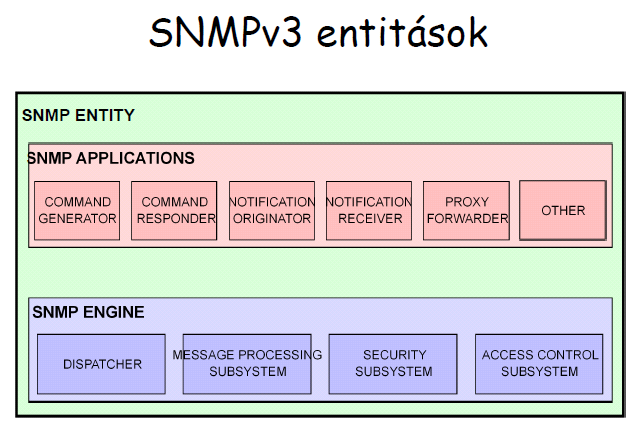
**Rajzolja le egy SNMPv3 ügynök (agent) moduljait! Egy-egy mondatban ismertess funkcióikat!**



**Kié az authoritative Engine ID egy SNMPv3 kommunikációban? Miért van rá szükség?**

The snmpEngineID of the authoritative SNMP engine involved in the exchange of this message. Thus, this value refers to the source for a Trap, Response, or Report, and to the destination for a Get, GetNext, GetBulk, Set, or Inform.

**Rajzolja le az SNMPv3 architektúrát! Részletezze az egyes modulok funkcióját!**



**Ismertesse a View-based Access Control Model-t! Milyen elemekből áll (értelmezze)?**

Read és write view-k tárolják csoportonként, hogy az adott csoport tagjai miket olvashatnak, ill. írhatnak.

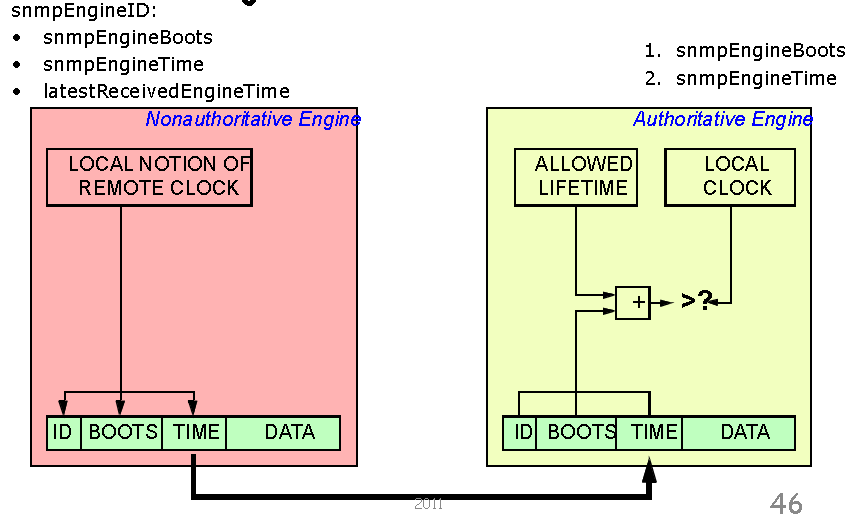
- Groups: felhasználói csoportok

- Security levels: üzenetbiztonság szintje

- Context: <a09-SNMPv3.pdf / 55. oldal>

**Hogyan működik a visszajátszás védelem (replay protection)? (milyen változókat tartanak nyilván? rajzoljon is!)**

Az authoritative engine-nek megfelelő ID, BOOTS, TIME értékek mennek át, ezzel időszinkront teremt.



**Hogyan építhet ki egy manager SNMPv3 biztonságos és titkosított kapcsolatot egy ismeretlen ágenssel? (Discovery)**

Két lépésben:

1. msgAuthoritativeEngineID meghatározása

non authenticated Request

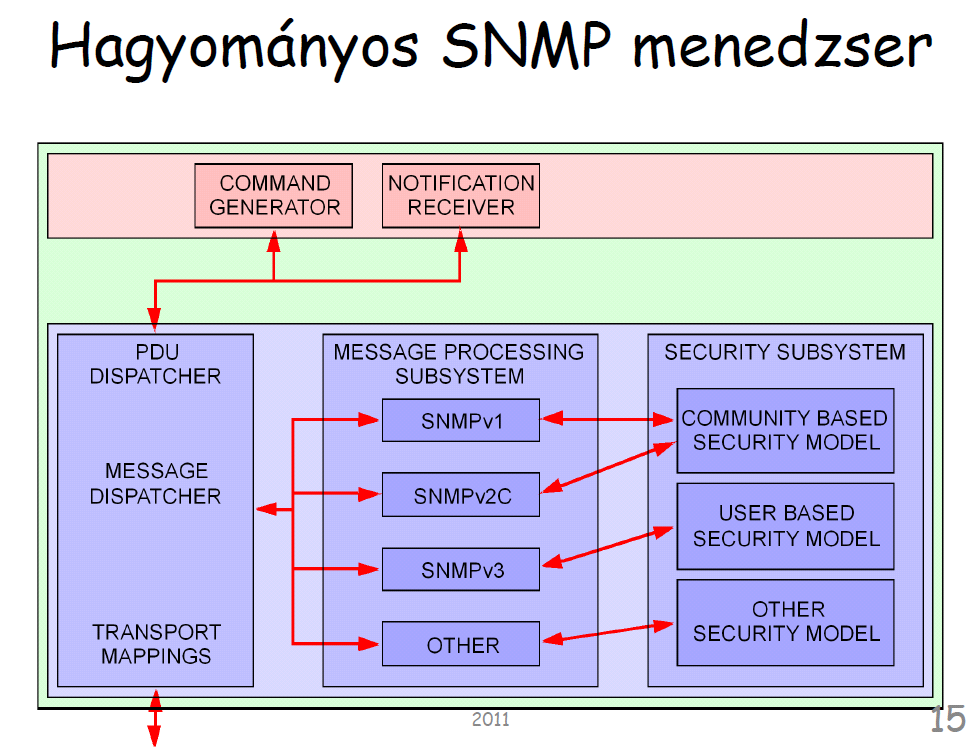
msgUserName= initial

msgAuthoritativeEngineID=null

2. establishing time synchronization

authenticated Request

**Rajzolja le egy SNMPv3 menedzser moduljait! Egy-egy mondatban ismertess funkcióikat!**



**10. MIB-Modellezés**

**Milyen osztályokba kategorizálhatjuk a menedzsment információkat? Hogyan viszonyulnak ezek egymáshoz?**

**Components** Menedzselt logikai és fizikai eszközök vagy szolgáltatások - motor

**Attributes** Jellemezően statikus tulajdonsága a modellezett objektumnak – Lóerő

**Actions** Vezérlés – ki, be, gyorsít

**Statistics** Hasznos információ a rendszer múltbéli állapotáról –max fordulatszám

**State** - Aktuális rendszer állapot RPM, olajnyomás, kibe

**Milyen megfontolásokat célszerű figyelembe venni egy MIB model készítésénél? Milyen egyszerű leképzésekkel lehet a modellből MIB-et készíteni?**

**Fentről-lefelé részletezve**

**Fizikai/logikai beágyazás?** (Mely fizikai eszköz tartalmazza, mely komm. rendszer része)

**Kardinalitás:** hány helyen fordul elő az adott elem a rendszerben.

**Statikus/dinamikus?**

**Csoportosítható/hierarchiába szervezhető?**

**Azonosítás módja:** véletlenszerű azonosítók/attribútumból származtatva

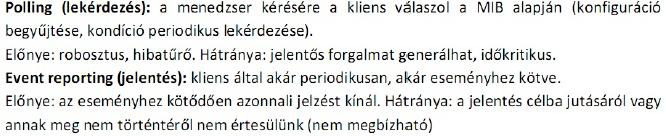
**Nincsenek explicit akciók:** Célokkal és lépésekkel oldjuk meg

**11. Patterns**

**Lehet-e lekérdezés alapú (polling) hibamenedzsmentet csinálni? Ha igen, mégis milyen indokok miatt használnak inkább jelentés alapút?**

**Lehet** pollinggal is, **de a jelentés alapú kevésbé terheli meg a hálózatot**, kevesebb fölösleges információt (állandó lekérdezések; még akkor is, ha sokáig semmi hiba nincs a hálózatban) küld a hálózaton. Ráadásul a polling „drága”, azért mert sok hálózati elemtől kell lekérdezgetni az állapotukat. Az event alapú további előnye hogy az esemény (hiba) bekövetkeztekor azonnal generálódik a jelentés, nem kell megvárni a következő lekérdezési periódust. (Ráadásul elképzelhető a legrosszabb eset, hogy a probléma mindig a lekérdezések között jelentkezik. Ekkor nyilván a megoldás a lekérdezések sűrítése, de ez csak fokozza a terhelést.)

**Hasonlítsa össze az esemény alapú és a lekérdezéses menedzsmentet előnyök és hátrányok szempontjából? Mikor melyiket használhatjuk? A gyakorlatban hogyan használjuk ezeket együttesen?**



**Milyen fundametális kommunikációs minták lehetnek a menedzser és az ágens között?**

Manager-initiated request and response, and agent-initiated events.

**Mik a problémák a tranzakció kezeléssel menedzsment rendszereknél? Mit csinálnak helyette? Mik ezen alternatívák előnyei és hátrányai?**

(**Probléma:** Hálózati management esetén a management műveletek elég nehézkesek, ha tranzakciókezelésről van szó, még akkor is, ha egyetlen eszközről van szó. A nehézséget az okozza, hogy számolni kell az időbeni késleltetéssel, a network-control protokollok zavarhatják egymást, a fizikai komponensek meghibásodhatnak, stb.)

Megoldás: A management alkalmazásoknak a fenti esetek előfordulását számításba kell venniük, ezért eléggé bonyolultak lehetnek.

● Verification (ellenőrzés): ellenőrzési lépésekre van szükség a konfigurációs műveletek alkalmazása előtt. Az ellenőrzés szintaktikai és szemantikai ellenőrzéseket is magába foglal. (Ezzel növelhető annak a valószínűsége, hogy a működés az elvártnak megfelel majd.)

● Validation (érvényesítés): a konfiguráció után ellenőrizni kell, hogy a műveletek hatásai az elvártaknak megfelel-e.)

//Sztem inkább az alábbira kíváncsiak, de nem tudtam egyértelműen eldönteni: Probléma: A rollback műveletek elvégzése a management alkalmazások felelőssége. A rollback sajnos nem (jól) kivitelezhető, mert elbukhat. Vannak olyan műveletek, amelyeket nem lehet visszavonni, mert már van visszavonhatatlan hatásuk.

Megoldás: Amikor a rollback nem lehetséges az a legjobb lépés, ha “roll forward”-ot hajtunk végre, ami azt jelenti, hogy a hálózatot egy jól definiált konfigurációs állapotba visszük. Előnyök: A hálózat konzisztenciáját helyre lehet állítani. Hátrányok: A nem az elvártnak megfelelő működés hatásai nem vonhatóak vissza.

**12. Functions**

**Milyen technikai indokok jöhetnek szóba a szolgáltatók flat-rate árazása mögött?**

Flat rate: Előre meghatározott teljes ár, amely független a teljesítés alakulásától, a megjelenésektől, stb

A szolgáltatóknak nem minden esetben van technológiai megoldásuk a használtsági adatok pontos kigyűjtéséhez, hogy az ügyfelek részére tételesen biztosítsák a számlázási adatokat.

**Mi a különbség a hibák szűrése és a hibák korrelációs elemzése között?**

**Szűrés esetén** csak bizonyos riasztások jutnak el a felhasználóig, az érintett alrendszer, szolgáltatás vagy súlyosság alapján. Szűrés az is, amikor a riasztások duplikálódását kívánjuk megakadályozni, és így a redundáns riasztásokat kiszűrni.

**A korrelációs elemzés** során több különböző riasztás helyett azokból egy következtetett összegző riasztást küldünk, mely értelmileg ugyanaz, mintha egyenként küldenénk el, de sokkal hatékonyabb módja a közlésnek.

**A TMN referencia modellben milyen menedzsment funkciók vannak? Mindegyiket definiálja 1 modatban!**

**Fault, Configuration, Accounting, Performance, Security**

**Hibamenedzsment:** monitorozás, hibákra való reakció. Általában Trouble Ticketing rendszerrel megsegítve.

**Konfigurációm.:** A hálózat beállításait, eszközeit követi. A változásokat dokumentálja. A menedzsment rendszer nyilvántartását szinkronizálja az eszközökben található beállításokkal. Más menedzsment rendszerek nagyon függnek a konfig.menedzsmenttől.

**Accountingm.:** A nyújtott kommunikációs szolgáltatások által szerzett bevétel menedzsmentje. Az erőforrások felhasználásának nyomonkövetése.

**Teljesítménym.:** A legfontosabb nem funkcionális tulajdonság. Pl. Átbocsátóképesség, késleltetés, megbízhatóság, jitter.

**Biztonságmen.:** A hálózat fenyegetésektől való védelme. A menedzsment biztonsága: csak az menedzseljen, akinek szabad. A biztonságosság menedzsmentje: védelem a vírusoktól, támadásoktól, behatolásdekektálás...

**Mi a különbség az Accountin és a Billing között?**

Az Accounting csak egy része a Billingnek. Ahhoz, hogy egy szolgáltatást számlázzunk, nem elég a körülményeket ismerni (mit, mikor, hogyan használtak), hanem szükségesek a tarifák ismeretei is, melyek segítségével eldönthetjük, hogy a könyvelt adatokat hogyan számlázzuk. Billing = Accounting + tarifa

**Milyen feladatai vannak a hiba menedzsmentnek? Milyen eszközöket használhatunk ennek segítésére?**

**Feladata a monitorozás és a megfelelő reakció az egyes hibákra.**

● **Network monitoring:** célja a hálózat működésének megfigyelése. Segítségével választ kaphatunk az egyes hibajelenségek okára.

● **Root cause (diagnosis):** Hibás működés, probléma esetén meg kell keresni a probléma eredetét.

● **Log history maintenance:** Az egyes esetek naplózása a későbbiekben hasznos lehet, ha hasonló hiba jelentkezik.

● **Trouble ticketing:** Tömör leírás a hiba jellemzésére. Kézről-kézre jár (eközben bővül a ticket), amíg valaki meg nem oldja a problémát.

● **Proactive fault management:** Megpróbáljuk megelőzni a hibát, vagy felkészülni annak bekövetkezésére.

● **Alarm handling**: Össze kell gyűjteni, meg kell jeleníteni és naplózni is kell az egyes riasztásokat.

**Miről szól a Performance menedzsment?**

Arról, hogy a hálózatot folyamatos monitoring segítségével úgy konfiguráljuk, hogy a lehető legjobb teljesítményt nyújtsa. Ezt pillanatnyi megfigyelések, és hosszú távú adatgyűjtés segítségével tudjuk megalapozni. Performance is the most important nonfunctional property.

**A konfiguráció menedzsment esetén milyen szinkronizációs megközelítéseket alkalmazhatunk? Melyik mely területeken terjedt jobban el? Miért?**

**Reconciliation(Egyeztetés)** abban az esetben történik, ha a hálózat állapotát tekintjük helyesnek, és ezért a MIB alapján frissítjük az adatbázisunkat. Ez az elterjedtebb megoldás, főleg céges környezetben, ugyanis leginkább az a fontos, hogy tudjuk, mi van a hálózatban.

**Reprovisioning** akkor történik, ha a menedzsment rendszerben tárolt információkat tekintjük helyesnek, és a hálózati eszközöket újrakonfiguráljuk azok alapján. Ez a megoldás főleg a telekommunikációs vállalatoknál jellemző, ugyanis ott a tervezett hálózatnak meg kell felelnie a kiépítettnek, ha nem, akkor az utóbbi a rossz. Előfordul, hogy egy cégen belül különböző hatáskörrel mindkét megközelítést alkalmazzák.

**Discrepancy reporting(Eltérés jelentés)** esetén a felhasználó dönti el minden egyes esetre, hogy melyik megoldást alkalmazza.

**Mi a különbség az auditing, discovery és autodiscovery feladatok között?**

Az **auditing** a rendszer állapotának kiolvasása, hogy meg tudjuk állapítani, hogy mi lett beállítva az eszközeinken.

A **discovery** alapjelentése annak felfedezése, hogy milyen eszközök vannak a hálózatban, azonban sokszor az auditing helyett használják tévesen ezt a kifejezést.

Az **autodiscovery** kifejezést szokták ezért néha a discovery funkciók leírására használni.

Másképp:

**auditing** – azzal foglalkozik, hogy ténylegesen mit konfiguráltak

**discovery** – kideríteni, hogy mi van a hálózatban

**autodiscovery** – automatikusan méri fel, hogy mi van a hálózatban (például: előre definiált időközönként)

**A Performance (teljesítmény) és Accounting (könyvelés) menedzsment területek hasonló adatokkal dolgoznak. Mi mégis a lényeges különbség közöttük az adatok gyűjtésére és felhasználásra vonatkozólag?**

**Accounting** esetén általában kevés paraméter lényeges, mivel az szolgáltatásonként meg van határozva, hogy mi alapján kell számlázni. Ezek **gyűjtése online**, és **felhasználásuk** többnyire **offline**, kivéve az előre fizetett szolgáltatások esetén.

**Performance** esetén minden egyes eszközről több különböző adatot szeretnénk megtudni, ezért rendkívül költséges lenne ezek folyamatos figyelése. Ehelyett az eszközök saját tárukban gyűjtik inkrementálisan az adatokat, majd a **lekérdezés egy alacsony forgalmú időszakban** (éjszaka) **történik**, és az adatokat egy célalkalmazás (numbercrunching) dolgozza fel.

Lényeges különbség:

A **Performance** menedzsment esetén nagy mennyiségű adat keletkezik. Az adatvesztés nem kritikus mindaddig, amíg nem befolyásolja jelentősen az ezekből készült statisztikát.

**Accounting** esetén kevesebb adat keletkezik, de nagyon fontos, hogy itt ne történjen adatvesztés, hiszen ez akár komoly bevételkieséssel is járhat.

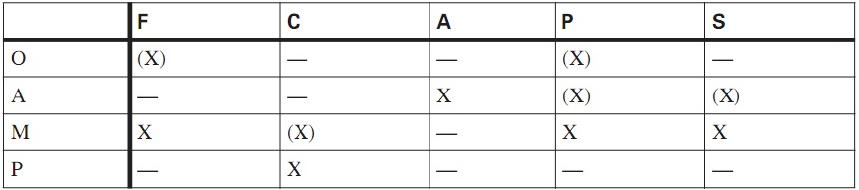
**Ismertesse az OAMP feladatokat! Viszonyítsa ezt az FCAPS funkciókhoz!**

**O:** a hálózat (és az általa nyújtott szolgáltatások) fennakadás nélküli működtetése. Beletartozik a hálózat folyamatos ellenőrzése (monitoring), hogy a hibákat a lehető leghamarabb észrevegyék, lehetőleg mielőtt egy felhasználónak problémája származna belőle. (részben Fault, Performance)

**A:** az erőforrások és azok kiosztásának számon tartása. Minden „házimunka” ide tartozik, mely ahhoz kell, hogy irányításunk alatt tartsuk a rendszert. (főleg Accounting, részben Performance, Security)

**M:** javítások és fejlesztések. Ide tartoznak a módosító és megelőző intézkedések is, a menedzselt hálózat jobbá tétele érdekében. (főleg Fault, Performance, Security, részben Configuration)

**P:** az erőforrások oly átrendezése, hogy egy adott szolgáltatást lehetővé tegyenek (Configuration)



**Adjon egy-egy példát hibamenedzsmentre az eszköz, a hálózati és a szolgáltatási rétegekre vonatkozólag!**

Érdemes sajátot kitalálni…

“Egy VoIP szolgáltatásra vonatkozólag ezt végig lehet gondolni:

**Üzleti réteg:** Help Desk (telefonos ügyfélszolgálat)

**Szolgáltatási réteg:** egyik call controller (SIP proxy) magas hívásblokkolást jelez.

**Hálózati:** adott útvonalon csomagvesztés, késleltetés vagy késleltetés ingadozás

**Eszköz szinten**: adott interfészeken hibás csomagok vétele”

**Mi az a TOM, milyen módon közelíti meg a hálózatmenedzsmentet? Hogyan viszonyulnak ezek a területek az FCAPS funkciókhoz?**

**Telecoms Operations Map**, amely a **Fulfillment, Assurance, Billing** megközelítést használja. A TOM középpontjában a menedzsment életciklus áll. Az OAMP helyett új megközelítéssel állt elő. A Fulfillment a Configuration feladatait látja el, a Billing az Accounting, az Assurance pedig a Fault, Performance és Security feladatait.

**Mi az a New Generation Operations Systems and Software (NGOSS)? Hogyan kapcsolódik ez az eTOM-hoz? Milyen működési területeket (folyamatokat) és funkciókat tartalmaz Level 0 szinten az eTOM?**

Wikipedia:

Frameworx, formerly known as NGOSS or "New Generation Operations Systems and Software" is the TeleManagement Forum’s programme to provide ways to help Communication Service Providers to manage their business. Frameworx includes a set of principles and technical deliverables.

The TM Forum Frameworx consists of four Frameworks, i.e.,

* Application Framework (sometimes referred to as the Telecom Application Map (TAM))
* **Business Process Framework (sometimes referred to as the enhanced Telecom Operations Map (eTOM))**
* Information Framework (sometimes referred to as the Shared Information/Data (SID) model)
* Integration Frameworks (which is developed in the TM Forum Integration Program (TIP))

**13. Other Management Protocols**

**Hogyan működik a CLI? Miért nem lehet könyen menedzsment rendszerekhez illeszteni?**

**Command Line Interface:** Humán bevitelre, és érzékelésre van tervezve, így nem olyan robosztus, és jól kereshető, mint azt egy menedzsment interfésztől elvárnánk. Nehézkes, vagy nem lehetséges a parancsok eredményeinek újra-felhasználása, struktúrába rendezése. A megjelenítés is problémát okoz, főleg nagy mennyiségű output adat esetén (screen scraping - MÁTRIX :) ) Kérdés-válasz párbeszédeket tesz csak lehetővé, az eseménykezelés alapból nem támogatott.

**Mi az a syslog? Hogyan működik? Hogyan illeszthető egy menedzsment rendszerbe? Előnyök és hátrányok?**

A syslog a UNIX rendszerekből származó mechanizmus, ami lehetővé teszi, hogy a menedzselt objetumok esemény üzeneteket adjanak ki, amik aztán logolva lesznek (minden üzenet egy új bejegyzés), és később menedzsment célokra felhasználhatóak. Előny, hogy így végigkövethetjük egy menedzselt eszköz összes tevékenységét, a CLI-hez hasonlóan “emberi nyelven.” Hátránya, hogy gyengén struktúrált (header-body) plaintext, így komoly feladat lehet a parse-olása (feldolgozása).

**Mi az a netconf? Mire használható? Milyen technológiát használ? Hogyan? Mi az a datastore?**

**Network Configuration Protocol.** Hálózati konfiguráció-menedzsmentet tesz lehetővé, monitorozást nem! XML alapú, RPC hívásokkal végez műveleteket MIB-eken, vagy azok részein.

A datastore egy fájlhoz hasonlóan tárolja az adott MIB-re vonatkozó konfigurációs beállításokat. Műveleteket datastore-okon végezhetünk.

**Mi az a netflow? Hogyan működik? Hogyan lehet a mintavételezés esetén becsülni a pontosságot? Milyen területeken lehet használni hálózatmenedzsmentben?**

IP folyam információk átvitelére használható protokoll. (Exporter -> Collector)

Használat alapú számlázáshoz használható (esetleg torrentezés szűrésére ☺).

**14. Scaling**

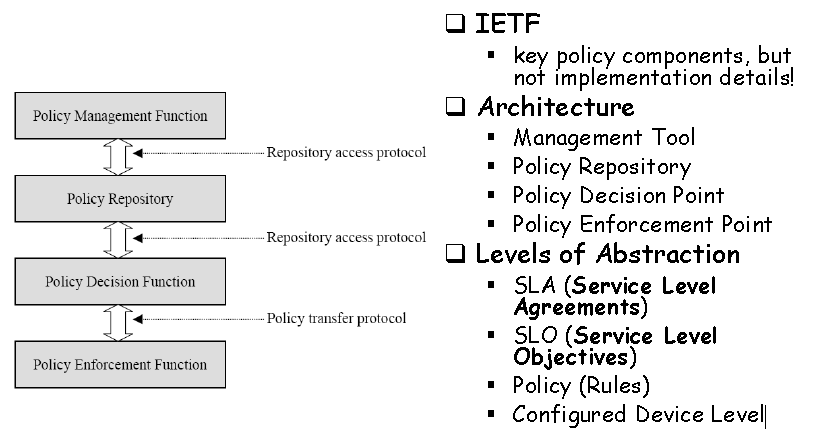
**Milyen menedzsment stílusokat ismer? Röviden ismertesse ezeket!**

**Delegációs menedzsment** Feladatok kiosztása a beosztottaknak. Pontosan meghatározzuk, hogy mit tegyenek, és lehetővé tesszük, hogy megtehessék.

**Célorientált menedzsment** Célokat tűzünk ki a beosztottak elé, és rájuk hagyjuk, hogyan érik el azokat.

**Kivételkezeléses (eszkalációs) menedzsment** A beosztottak a felelősök/döntéshozók, de amennyiben valami szokatlan történik, és eszkaláció szükséges, akkor mi is belépünk a megoldási folyamatba.

**Ismertesse az IETF-es Policy keretrendszer komponenseit, köztük használt protokollokat!**



**Milyen rétegekben lehet mediálni menedzsment rendszerekben? Mutassa be ezeket példákon keresztül!**

- **Átviteli réteg** UDP → SSH

A netconf SSH UDP keretben nem érkezett meg, a kapcsolat nem épül ki, amin UDP keretek tunneleznének.

- **Menedzsment protokoll réteg** CLI vs. SNMP get , syslog vs. SNMP trap

Szabályok, és template-ek használatával nem mosható össze a két mediált mgmt protokoll.

- **Menedzsment információ réteg** SNMP V3 trap PDU mappelése syslog üzenetté (szintaxis)., Egyedi fordítási szabályok(szemantika).

Az a gond, hogy a feldolgozás sajnos a menedzsment rendszerre hárul.

**Miért készítenek hierarchikus menedzsment rendszereket? Milyen módszerekkel lehet hierarchikusan kiszervezni funkciókat?**

Lokálisan releváns, kisebb sávszélesség igényű, kompaktabb adathalmazokkal dolgozó rendszerek hozhatóak így létre, valamint előny, hogy a rétegek csak az alattuk, ill felettük levő rétegekkel kommunikálnak, így a menedzsment információk is hierarchizálva lesznek. A rétegek kialakításával levehetjük a menedzsment szoftverről az egyszerű, ámde számítás-, és/vagy sávszélesség-igényes feladatokat, amiket így az alsóbb rétegek szolgáltatnak majd számára, strukturált információkként.

**Hogyan kezelhető a menedzsment rendszer felépítésének komplexitása?**

**A horizontális (taszkok), és vertikális (rétegek) mentén**. Taszkok modulokra bontása és egymásra mappelése a bulid komplexitás (menedzsment rendszert fenn kell tartani, fejleszteni, bővíteni) miatt. Rétegek kialakítása felelősségek és megvalósítandó funkciók igényei alapján (számításigényes, időkritikus, tárhelyigényes stb.) a runtime komplexitás (lépést kell tartani a hálózat növekedésével).

**15. Service Level Management**

**Milyen négy pilléren (elven) "nyugszik" a szolgáltatás minőség (QoS) biztosításának feltétele csomagkapcsolt IP hálózatokban? Rajzoljon és magyarázza el ezeket az elveket!**

Pillars of QoS:

Packet classification

Isolating: scheduling and policing

High resource utilization

Call admission

* Packet classification: packet marking needed for router to distinguish between different classes; and new router policy to treat packets accordingly
* Isolation(scheduling and policing): provide protection (isolation) for one class from others
* High resource utilization: While providing isolation, it is desirable to use resources as efficiently as possible
* Call admission: flow declares its needs, network may block call (e.g., busy signal) if it cannot meet needs

**Milyen szempontokat kell figyelembe vennie egy szolgáltatónak, ha SLA megfelelésnek megfelelően szeretné a hálózatát és szolgáltatásait működtetni?**

**planning:**

- Reserve resources for SLAs: dimensioning problem

- Oversubscription Risk

**monitoring parameters:**

- source: mgmt information, SNMP MIB, syslog, netflow, ipfic stb.

- Passive and Active measurements

**Preventing Failures:**

- Service level forecasts, good tools: Threshold-crossing alerts

**Keeping Record:**

- important as a proof of service

- Sometimes service level report are part of the service package

- statistics témakör…