Az adatátvitel megbízhatósága járműkövető rendszerekben

Aradi Szilárd*. Bécsi Tamás**

**BME Közlekedésautomatikai Tanszék*
Budapest (Tel: (1)463-1044; e-mail: aradi.szilard@mail.bme.hu).
*** BME Közlekedésautomatikai Tanszék*
Budapest (Tel: (1)463-1044; e-mail: becsi.tamas@mail.bme.hu).

Absztrakt: A GPS és GSM technológiák fejlődésének, valamint a beruházási és kommunikációs költségek csökkenésének köszönhetően egyre elterjedtebbek az on-line járműkövető rendszerek. Az ilyen jellegű megoldások mind funkcionalitásukban, mind az átvitt adatok mennyiségében folyamatosan bővülnek, ami a rendszerelemek komplexitását nagyban növeli. Jelen cikkünkben a járműkövető rendszereket az adatátvitel megbízhatósága szempontjából vizsgáljuk, ami általános célú rendszereknél a használhatóságot és a szolgáltatás minőségi szintjét növeli, míg biztonsági igényű rendszereknél (pl.: vasúti területen) az alapvető tulajdonságok közé tartozik.

1. BEVEZETÉS

A mikroelektronika és a mobil távközlés gyors ütemű fejlődésének köszönhetően a kilencvenes évek végére megvalósíthatóvá vált a járművek egyre szélesebb körű fedélzeti diagnosztikája és műholdas nyomon követése. A gazdasági igény a járműkövető rendszerekre a személy- és áruszállításban jelentkező növekvő versenyhelyzet hatására erősödött meg, elsősorban a közúti közlekedés területén.

Az on-line járműkövető rendszerek terjedését nagyban segítette a kommunikációs költségek folyamatos csökkenése, valamint az adatátviteli sebesség növekedése.

Ezeknek a rendszereknek az alkalmazása nagyon sok olyan előnnyel jár, amelyek megteremtik a létjogosultságát mind a közúti, mind pedig a vasúti közlekedésben. Ezek az előnyök a következők:

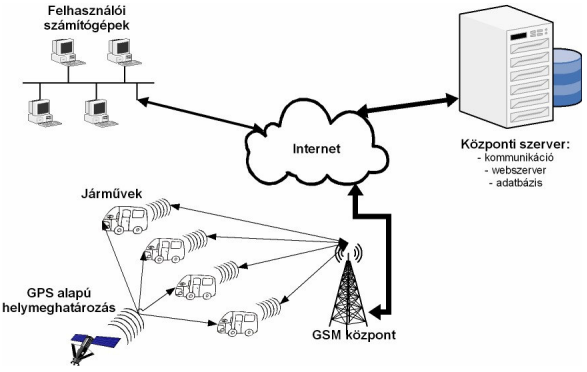
- nagyobb a szállítási biztonság,
- dinamikus fuvarszervezés elősegítése,
- a járművek műszaki állapotának folyamatos követése,
- könnyebb dokumentáció,
- a teljesítmény alapú bérezés elősegítése,
- a közlekedésbiztonság javítása,
- a szállításbiztonság javítása,
- fokozottabb környezetvédelem.

Az megbízhatósági és rendelkezésre állási igények, a járműkövető rendszerekkel szemben, egyre nagyobbak. Ez a cikk egy általános on-line járműkövető rendszer lehetséges felépítését mutatja be, külön hangsúlyt fektetve az adatátvitel megbízhatóságára és az on-line rendelkezésre állásra.

2. RENDSZER

Az on-line flottamenedzsment rendszerek általános felépítését az 1. ábra szemlélteti. A rendszer három fő eleme:

- a fedélzeti egység,
- a központi szerver,
- a felhasználói számítógépek.



1. ábra: Az on-line járműkövető rendszerek általános felépítése

A rendszer működése a következő. A járművön lévő fedélzeti egységek mérik a jármű működési paramétereit (kapcsolók, relék állapota, energiafelhasználás, motorparaméterek stb.), és pozícióját (GPS alapú helymeghatározás segítségével), valamint tárolják a járművezető által megadott adatokat (a szállított áru adatai, az aktuálisan végzett tevékenység megnevezése stb.). Ezeket az értékeket előre definiált események bekövetkeztekor (vészjelzés, hirtelen gázolajszint csökkenés stb.), illetve előre definiált időközönként elküldik egy központi szervernek.

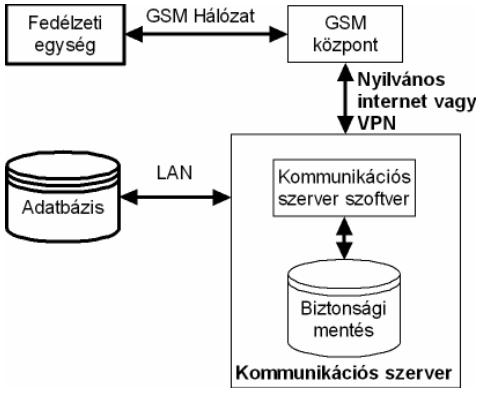
A fedélzeti egységek mobilhálózaton keresztül kommunikálnak a központi szerverrel. A beérkezett adatok ellenőrzésre kerülnek, és egy adatbázisban tárolódnak. Amennyiben szükséges, a központi szerver riasztást küldhet egy adott e-mail címre vagy akár mobiltelefonra is. Ebben a struktúrában megoldható a szerverről a jármű felé történő kommunikáció is. Ennek segítségével a beérkezett adatcsomagokat vissza lehet igazolni, szöveges üzenet küldhető a vezető számára, illetve beállíthatóak a fedélzeti egység működési paraméterei.

Folyamatosan (on-line) követhetőek és figyelhetőek a mozdonyok, valamint a központban tárolt adatok utólagos (off-line) kiértékelésével az üzemeltetés paramétereit (szállítási teljesítmények, energiafelhasználási adatok, járművezetők tevékenységei, munkaideje stb.) követhetjük nyomon.

3. ADATÁTVITELI ÚT

Az on-line járműkövető rendszereknél az adatokat nagy megbízhatósággal és integritással kell eljuttatni a járműről egy központi adatbázisba.

Az adatátviteli kiinduló pontja (2. ábra) a járműfedélzeti egység, amely egy modem segítségével GSM (Global System for Mobile communications) hálózaton keresztül kapcsolódik a kommunikációs szerverhez. A szerver fogadja az adatokat, majd a megfelelő ellenőrzések és konverziók után, egy adatbázisba írja az információkat.



2. ábra: Az adatátviteli út felépítése

3.1 Fedélzeti berendezés

A fedélzeti berendezés egy mikroszámítógépre épülő adatgyűjtő, tároló, és továbbító eszköz. A fedélzeti egység felépítésénél fontos szempont a robusztusság (EMC védelem, rázkódásvédelem stb.) és a modularitás. A kommunikációt egy GSM modem valósítja meg, amely szintén rendelkezhet saját mikroszámítógépes erőforrásokkal (CPU, RAM, flash memória), így képes lehet a teljes kommunikációs protokollt (lásd 4.fejezet) kezelni.

3.2 GSM hálózat

A járműkövető rendszerek jelenleg a nyilvános GSM hálózatot használják adatátviteli célokra. Erre három lehetőség használható járműkövető rendszerek esetén:

- SMS alapú,
- adatkapcsolt és
- csomagkapcsolt

adatátviteli technológia. Napjainkban a csomagkapcsolt adatátvitel az egyeduralkodó. Ezek közül a legelterjedtebb a GPRS (General Packet Radio Service), illetve a nagyobb adatátviteli sebességet biztosító EGPRS (Enhanced GPRS). A 3G hálózatokban lehetőség van még az UMTS és a HSDPA használatára, azonban az alacsony lefedettség és az eszközök magas ára miatt a technológia még nem terjedt el a járműkövető rendszerekben.

A csomagkapcsolt technológia előnyei a következők:

- állandó kapcsolat,
- nagyobb adatátviteli sebesség,
- adatmennyiség alapú számlázás,
- alacsony költségek.

Az SMS alapú adatküldés használata megfelelő lehet biztonsági tartaléknak a GPRS szolgáltatás hiánya esetén, valamint speciális adatok (pl.: riasztás) közvetlenül, mobiltelefonra történő küldésére.

Azonban a fent felsorolt indokok miatt a GPRS a legmegfelelőbb technológia, mely a vonalkapcsolt GSM szabványon alapuló csomagkapcsolt hordozószolgálat. A GPRS rádió interfésze a GSM szabványra épül úgy, hogy bevezetésével a GSM vonalkapcsolt technológia változatlan marad. Éppen ezért a GPRS a GSM rádiós interfészen is használt GMSK (Gaussian Minimum Shift Keying) modulációs eljárást alkalmazza. A GPRS rádiós interfészen az adatcsomagokat rádiós blokkonként továbbítják, minden blokk 456 bitet tartalmaz. A rádiós erőforrások a blokkokhoz kerülnek kijelölésre, nem pedig a forgalmi csatornákhoz, mint a GSM-ben. Ez sokkal hatékonyabb kihasználást tesz lehetővé, a GPRS ugyanis dinamikusan csak akkor jelöl ki rádiós erőforrást, ha valóban van adatforgalom. Ezáltal több felhasználó osztozik ugyanazon a fizikai csatornán, és a cellában a GSM és a GPRS felhasználók közösen férnek hozzá a rádiós erőforrásokhoz. A GPRS lehetővé teszi, hogy egy mobil állomás egy TDMA (Time Division Multiple Access - időosztásos többszörös hozzáférés) keret több időrésben is adhat (multi-slot operation), továbbá az adási (uplink) és vételi (downlink) irány külön kezelhető, ezzel a GPRS támogatja az aszimmetrikus forgalmat.

A GPRS három új kódolási eljárást vezet be, melyek nagyobb átviteli sebességet tesznek lehetővé egy időrésben, mint a hagyományos GSM azonban kisebb hibavédelmet biztosítanak. A GPRS által biztosított elvi maximális adatsebesség 171,2 kbit/s a legkisebb adatvédelmet biztosító CS-4 csatornakódolási eljárás mellett, 8 időrés összefogásával. A gyakor-

latban hálózati és készülék oldalról is a CS-2 kódolás érhető el, és 3 időrés fogható össze, így a rádiócsatorna 40,2 kbit/s sebességet biztosít, ami az alkalmazói réteg szintjén 30-33 kbit/s sebességre csökken az alkalmazástól függően.

Az EDGE (Enhanced Data rates for GSM Evolution) más modulációs eljárásra épül, mint a GSM, bevezeti a 8-PSK (Phase Shift Keying) modulációt, mellyel nagyobb átviteli sebesség érhető el. Az EDGE keretében a továbbfejlesztett EGPRS alkalmazói szinten legfeljebb 220 kbit/s sebesség elérését teszi lehetővé.

Ezek az adatátviteli sebességek a járműkövető rendszerekhez – megfelelően megtervezett protokoll esetén – elegendőek, így nem szükséges a 3G hálózat használata.

3.3 Kommunikációs szerver

A kommunikációs szerver a járműkövető rendszerek központi eleme.

A szerver fő feladatai a következők:

- adatok fogadása a járművekről,
- adatok ellenőrzése,
- adatok nyugtázása,
- járművezető azonosítása,
- adatok adatbázisba írása,
- riasztás küldése, amennyiben szükséges,
- fedélzeti egységek működési paramétereinek beállítása,
- távdiagnosztika,
- szoftverfrissítés.

Az adatrekordok fogadása IP (Internet Protokoll) alapon történik. A szállítási réteg lehet TCP vagy UDP. A kommunikációhoz nem érdemes magasabb szintű protokollt használni (pl.: FTP), mert a sok fájlművelet nagyszámú kliens (több ezer jármű) esetén lelassíthatja a rendszert. Érdemesebb egy saját IP alapú kommunikációs protokollt (4. fejezet) kifejleszteni.

3.4 Adatbázis

A járműről érkező adatok tárolására a legalkalmasabb egy megfelelően megtervezett relációs adatbázis. Így az utólagos kiértékelések elkészítése sokkal hatékonyabb lehet. Az adatbázisnak tárolnia kell a beékezett adatokon kívül, a járművezetők azonosításához szükséges adatokat is. Az adatbázis szerver kialakításánál nagy gondot kell fordítani az adatbiztonságra és a rendelkezésre állásra.

4. PROTOKOLL

Az adatátviteli kialakításánál a 3.2 fejezetben részletezett előnyök miatt a GPRS technológiát választjuk, emiatt az IP (Internet Protokoll) alapú kommunikáció adott. (A továbbiakban IP rövidítés alatt IPv4 protokollt értjük.) Ebben az esetben az OSI modell szerinti fizikai, adatkapcsolati és hálózati rétegek adottak a jármű és a szerver oldalán is. Az első három réteg megvalósítását a GSM modem, a GSM hálózat és a szerver hardvereszközei, operációs rendszere és szoftverei valósítják meg. A szállítási réteg protokolljánál a két legelterjedtebb megoldási lehetőséget hasonlítjuk össze.

4.1 User Datagram Protocol (UDP)

Az UDP egy kicsi, egyszerű, üzenet-központú szállítási protokoll, melyet az IETF (Internet Engineering Task Force) RFC 768-as szabványa tartalmaz. Az UDP egy nagyon egyszerű interfészt biztosít a hálózati réteg és a felsőbb rétegek (viszonylati és alkalmazási réteg) között.

1. táblázat: Az UDP csomag felépítése

+	Bit 0 - 15		16 - 31
0	Forrás cím		
32	Cél cím		
64	Nullák	Protokoll	UDP hossz
96	Forrás port		Cél port
128	Hossz		Ellenőrző összeg
160	Adat…		

Az UDP csomagok felépítése az 1. táblázatban látható. A táblázatban szürke hátérrel az IP fejlécből átemelt mezők szerepelnek. A csomag elemei a következők:

- **Forrás port:** a küldő portszámát adja meg, kitöltése nem kötelező.
- **Cél port:** a fogadó portszámát adja meg, kitöltése kötelező.
- **Hossz:** A csomag teljes hossza: fejléc + adat. Az elméleti maximum 65 535 byte, azonban a felette lévő IP protokoll miatt a gyakorlatban 65 507 byte lehet a maximum méret.
- **Ellenőrző összeg:** Egy 16 bites ellenőrző összeg, amely adatvédelmi célokat szolgál. Az ellenőrző összeg – az RFC 768 szabvány szerint – a 16 bites egyes komplemense a pszeudo fejléc, az UDP fejléc és az adat, egyes komplemensű összegének. Ha szükséges, nullákkal kell feltölteni az adatsor végét, hogy a teljes méret a 16 bit többszöröse legyen. Pszeudo fejlécnek nevezik az IP fejléc azon részét (lásd 1. táblázat szürke része), amelyet az UDP ellenőrző összegbe bele kell számítani. Ez tartalmazza a küldő és a fogadó IP címét, így elkerülhető a hibás címre küldés.

- **Adat:** a küldendő adat. Maximális mérete 65 499 byte lehet. A fenti négy mezőt összefoglalóan UDP fejlécnek nevezik.

Az UDP protokoll használata esetén nem alakul ki viszonylat a forrás a cél között, a csomagokat előkészítés nélkül adják fel a kommunikáció során. Ezért az UDP-t a kapcsolatnélküli protokollok közé sorolják. Az UDP nem garantálja a felsőbb rétegek számára az üzenet megérkezését és sorrendjét, továbbá a küldő nem kap információt az általa elküldött üzenet státuszáról. Az adatcsomagok a továbbítás során elveszhetnek. A protokoll használata esetén a sorrendet és a megbízhatóságot a felsőbb rétegeknek kell biztosítaniuk. Kis mérete és egyszerűsége miatt gyors és hatékony, ezért időkritikus alkalmazásokban (pl.: IP alapú hangátvitel) jól használható.

4.2 Transmission Control Protocol (TCP)

A TCP a legelterjedtebb IP alapú szállítási protokoll. A TCP-vel együtt fejlesztették az Internet Protokollt is. A TCP megbízható adatátvitelt biztosít a protokollt használó alkalmazási rétegnek.

2. táblázat: Az TCP csomag felépítése

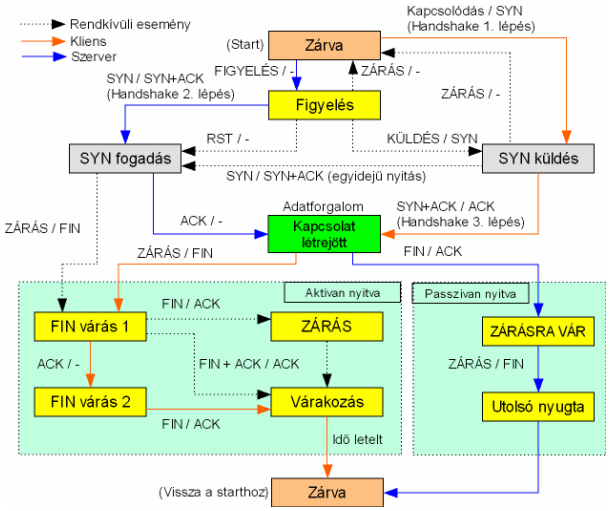
+	Bit 0 - 3	4 -7	8 – 15	16 - 31
0	Forrás cím			
32	Cél cím			
64	Nullák		Protokoll	TCP hossz
96	Forrás port			Cél port
128	Sorszám			
160	Nyugta sorszám			
192	Adateltolás	Foglalt	Jelző bitek	Ablak
224	Ellenőrző összeg			Sürgősségi mutató
156	Opciók			
156/ 288+	Adat			

AZ UDP-vel ellentétben - amely azonnal küldi a csomagokat – a TCP az adatküldés előtt létrehoz egy kapcsolatot (TCP socket) a kommunikáló felek között, így a viszonylati réteget is biztosítja. A teljes kommunikációs folyamat (3. ábra) három fázisból áll:

- kapcsolat létrehozása,
- adatátvitel,
- kapcsolat lezárása.

A kapcsolat létrehozásához a TCP egy ún. 3-utas „handshake”-et használ. Mielőtt egy kliens megpróbál csatlakozni egy szerverhez, a szervernek meg kell nyitnia egy portot a csatlakozások fogadásához: ez az ún. passzív nyitás. Nyitott port esetén a kliens kezdeményezheti a kapcsolat létrehozását a következő módon:

- A kliens egy szinkronizációs csomagot (SYN) küld a szervernek.
- A szerver erre válaszol egy szinkronizációs nyugtával (SYN-ACK).
- Végül a kliens is visszaküld egy nyugtát (ACK) a szervernek.



3. ábra: Egyszerűsített TCP állapotdiagram

Az adatküldés során a következő fontos tulajdonságokkal rendelkezik a TCP.

- **Rendezett adatátvitel:** Amennyiben a csomagok nem az eredeti sorrendben érkeznek meg a fogadó félhez, úgy a TCP automatikusan sorbarendezi azokat, a sorszám mezőnek megfelelően.
- **Elveszett csomagok újraküldése:** Bármely adatcsomagot, amelyre a küldő fél nem kap nyugtát, a TCP automatikusan újraküldi.
- **Duplán küldött csomagok eldobása:** Amennyiben ugyanazon adatcsomag többször kerül elküldésre, a felesleges csomagokat a protokoll eldobja.
- **Forgalomvezérlés:** A TCP forgalomvezérlési protokollja gondoskodik arról, hogy amennyiben a küldő túl gyorsan küldi az adatokat, fogadó fél akkor is megbízhatóan képes legyen azokat fogadni és feldolgozni. Így kiküszöböli a sebességkülönbséget a kommunikáló felek között, ezért az nem vezet adatvesztéshez.
- **Erős adatvédelem:** A fent felsorolt tulajdonságok mind a hibamentes adatátvitelt szolgálják. Ezeken felül a TCP rendelkezik egy 16 bites ellenőrző összeggel. A számítás módja ugyanaz, mint az UDP esetén, azonban a számításba beleveszi az ellenőrző

összeg mezőjét (2. táblázat) is nullákkal feltöltve. Ez az ellenőrző összeg manapság már gyengének mondható, azonban az adatkapcsolati réteg (pl.: Ethernet) is - a legtöbb esetben - tartalmaz adatvédelmet, így növelvén a végpontok közötti adatátvitel megbízhatóságát.

A kapcsolat lezárása 3- vagy 4-utas „handshake”-kel történhet. A 4-utas változatnál az egyik végpont küld egy lezáró csomagot (FIN), amire a másik egy nyugtával (ACK) válaszol, majd ezt megismétlik, tehát a lezárás egy pár FIN és ACK csomag küldéséből áll. A kapcsolat lehet félig nyitott állapotban is, amikor az egyik fél már lezárta a folyamatát, azonban a másik még nem. Ekkor a lezárt fél már nem tud adatot küldeni, azonban a másik fél még igen, de már nyugtát nem fog rá kapni. A 3-utas változatnál az első végpont küld egy FIN csomagot, a másik pedig a egy FIN&ACK csomaggal válaszol (két lépést összevon a 4-utashoz képest), majd az első egy ACK csomagot küld vissza. Ez a lezárási mód a leggyakrabban használatos.

Amint a fent leírtakból látszik, a két szállítási protokoll nagyban eltér egymástól. Az UDP valós idejű alkalmazásoknál jól használható, mivel egyszerű és gyors, kapcsolat nélküli protokoll. Azonban a kommunikáció megbízhatóságról a magasabb OSI rétegekben kell gondoskodni. A TCP egy hosszú évek alatt kifejlesztett protokoll, amely folyamatos, közvetlen kapcsolatot biztosít a két végpont között. Megbízhatósága magas fokú, azonban felépítése és működése bonyolult és összetett, emiatt az implementálása nehezebb, valamint az adatátvitel lassabb a nagyobb méretű fejlécek miatt. Ezek a hátrányok azonban ma már nem jelentenek akadályt, mivel a PC-s és szerver operációs rendszerek is széleskörűen támogatják a használatát. Használata a járműoldalon is leegyszerűsödött, hiszen egyre elterjedtebbek azok a GSM modemek, amelyek tartalmaznak ún. „TCP stack”-et, így a kommunikáció egészen a szállítási rétegig implementálva van az eszközben.

4.3 A megjelenési réteg protokollja

A viszonylati réteg feletti kommunikációs lehetőségeket az adatstruktúra alapján két fő csoportra oszthatjuk:

- Egyszerű (bájtos) adatstruktúra
- Leíró nyelv használata

Az első esetben a járművön mért adatokat nyers formában, előre definiált sorrendben továbbítjuk a szerver felé. Az adatrekord az elején és végén speciális - máshol nem használt - bájtokat tartalmaz, ezzel jelölvén egy adat kezdetét és végét.

3. táblázat: Példa az egyszerű adatstruktúrára

	1. bájt	**2.**	**3-4.**	**n.**	**(n+1).**
Jelentés	Start	ID	Fuel	…	End
Kódolás	ASCII	BCD	BCD	…	ASCII
Mértéke.	-	-	liter	…	-
Érték	$ (0x24)	12 (0x12)	0230 (0x0230)	…	* (0x2A)

A 3. táblázatban láthatunk egy példát arra, hogy milyen módon lehet létrehozni egy egyszerű adatstruktúrát. A mezők jelentését, kódolását és mértékegységét mindenképp szükséges definiálni, hogy az adatok értelmezhetőek legyenek. Az e fajta adatábrázolás előnye, hogy nagyon tömör, így csökkenti az átvitt adatmennyiséget. Hátránya, hogy nehezen bővíthető, továbbá hibakeresés esetén nehezen olvasható.

A másik lehetőség egy leíró nyelv használata. Az egyik legelterjedtebb, szabványosított leíró nyelv az XML (Extensible Markup Language), ezért a továbbiakban ezen keresztül kerül bemutatásra ez a módszer.

Az XML a W3C (World Wide Web Consortium) által ajánlott általános célú leíró nyelv, speciális célú leíró nyelvek létrehozására. Az SGML (Standard Generalized Markup Language) egyszerűsített részhalmaza, mely különböző adattípusok leírására képes. Az XML-en alapuló nyelvek formális módon vannak leírva, így lehetővé téve a programok számára a dokumentumok módosítását és validálását a formátum előzetes ismerete nélkül. Az XML azon tulajdonságai, melyek alkalmassá teszik adattovábbításra, a következők.

- Mind ember, mind gép számára olvasható formátum.
- Képes a legtöbb számítástudományi adatstruktúra ábrázolására (rekord, lista, fa...).
- Támogatja a Unicode-ot, ami nagyon bőséges karakterkészletet állít rendelkezésre.
- Öndokumentáló formátum, amely struktúra- és mezőneveket ír le speciális értékekkel együtt.
- Szigorú szintaktikus és elemzési követelményeket támaszt, ami biztosítja, hogy a szükséges elemzési algoritmus egyszerű, hatékony és ellentmondásmentes maradjon.
- Platform független, így viszonylag érzéketlen a technológiai változásokkal szemben.
- Egyszerű szöveg formátumban valósul meg, licenszektől és korlátozásoktól mentesen.
- Széles tapasztalat és eszközkészlet áll rendelkezésre az XML alapú fejlesztésekhez.

A szigorú szintaktikus és elemzési követelmények lehetőséget biztosítanak arra, hogy a dokumentum helyességét és érvényességét ellenőrizzük. Egy helyesen formázott XML dokumentum megfelel minden szintaxis szabálynak. Az a dokumentum, ami nem helyesen formázott, nem tekinthető XML-nek. Egy érvényes dokumentum olyan adatot tárol, ami megfelel a felhasználó által definiált tartalmi szabálynak, ami leírja a helyes adat értékeket és helyeket. Ennek megadására DTD-t (Document Type Definition) vagy XSD-t (XML Schema Definition) használhatunk. A DTD az XML 1.0 szabvány része, így mindenhol támogatott. Az XSD egy

újabb keletű XML séma nyelv, amit a W3C a DTD utódaként definiált. Az XSD lényegesen többre képes a DTD-nél az XML nyelvek leírása terén. Sokoldalú adattípus rendszert használ, ami részletesebb megkötéseket tesz lehetővé az XML dokumentum logikai szintjén, de ezért sokkal robusztusabb érvényesítő keretrendszert követel meg. Ráadásul az XSD XML-alapú formátumon alapul, minek következtében szokványos XML eszközöket lehet használni a létrehozásához és feldolgozásához, bár az implementációk sokkal többet kívánnak, mint az egyszerű XML olvasási képesség. A fenti okok miatt a továbbiakban az XSD használatát mutatjuk be. Az XML sémák készítése jól támogatott, sok készen kapható szoftvereszköz áll rendelkezésre. A következőkben egy XSD részletet mutatunk be, amely egy jármű GPS adatait írja le.

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema elementFormDefault="qualified" id="MFB"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
 <xs:complexType name="GPS">
  <xs:sequence>
   <xs:element minOccurs="1" maxOccurs="1" name="status">
    <xs:simpleType>
     <xs:restriction base="xs:string">
      <xs:length value="1" />
     </xs:restriction>
    </xs:simpleType>
   </xs:element>
   <xs:element minOccurs="1" maxOccurs="1" name="hossz"
type="xs:double" />
   <xs:element minOccurs="1" maxOccurs="1" name="szelesseg"
type="xs:double" />
   <xs:element minOccurs="1" maxOccurs="1"
name="iranyszog" type="xs:integer" />
   <xs:element minOccurs="1" maxOccurs="1" name="sebesseg"
type="xs:integer" />
  </xs:sequence>
 </xs:complexType>
```

A „GPS” mezőt komplex adattípusként írjuk le, amely tartalmazza a „status”, a „hossz”, a „szelesseg”, az „iranyszog”, és a „sebesseg” elemeket. Minden elemhez megadtuk az adatípust (byte, double, integer, vagy string), valamint a minimális és maximális előfordulások számát. Mivel mindkettőnek „1”-es értéket adtunk, ezért ezeket az elemeket kötelezően tartalmaznia kell az adattípusnak egy – a séma alapján készített – XML-nek. A „status” elemnek megadtuk a hosszát is, így csak egyetlen karakter fogadható el értékként. Egy példa a séma alapján készített XML-re.

```
<gps>
  <status>A</status>
  <hossz>47.480101</hossz>
  <szelesseg>19.057557</szelesseg>
  <iranyszog>270</iranyszog>
  <sebesseg>35</sebesseg>
</gps>
```

Látható, hogy az XML szabvány definiálja a szintaktikát, saját XML séma készítésével pedig, definiálni lehet az adatok érvényességét is. A kommunikáció során a séma egyik oldalról támogatja a megfelelő XML felépítését, míg a másik oldalról a fogadott adatok validálására használható. Amennyiben a séma definiálását, és az adatok validálását következetesen végezzük, valamint – a lehetőségek szerint – minél jobban szigorítjuk, úgy az érvénytelen adatok már az adatfeldolgozás elején kiszűrhetők, és figyelmen kívül hagyhatók.

4.4 Az alkalmazási réteg

Alkalmazási réteget csak a szerver oldalon van értelme definiálni. A réteg feladata a strukturált, jól kereshető, könnyen hozzáférhető, és lehetőleg platformfüggetlen adattárolás. Erre a legalkalmasabb egy SQL (Structured Query Language) alapú, relációs adatbázis. A kommunikációs szerver az adatbázisba helyezi el az adatokat, valamint innen kérdezi le a járművezetők azonosításához szükséges adatokat. Az adatbázis szerver futhat a kommunikációs szerverrel egy hardveren, vagy egy – helyi hálózaton elérhető – különálló adatbázisszerveren. A legfontosabb és a legnagyobb terheléssel járó feladat a járműről érkező adatok beillesztése. Itt kétféle stratégia választható. Az első esetben a kommunikációs szerver minden járműhöz különálló adatbázis kapcsolatot hoz létre. Ebben az esetben minden egyes érkező adat külön kerül beillesztésre az adatbázisba. Nagy számú kliens esetén a sok adatbázis kapcsolat és a sok különálló parancs nagyon leterhelheti az adatbázisszervert. A másik megoldás során a kommunikációs szerver csak egy kapcsolatot létesít a szerverrel és minden adatbeillesztés és lekérés ezen keresztül folyik. Fejlett adatbázis szerverek képesek egy parancs futtatásával több sort is beilleszteni, nagy sebességgel, így nagy számú kliens esetén alacsony terheléssel futhat az adatbázis. Ennek a módszernek az implementálása nagyobb körültekintést igényel, azonban sok járműből (százas nagyságrendű járműszám) álló flotta esetén mindenképpen ezt szükséges alkalmazni.

Összefoglalva a fentieket az adatátviteli rendszer a következőképpen épülhet fel az OSI-modell szerint.

4. táblázat: Az adatátviteli rendszer felépítése

OSI modell	Használt technológia
Fizikai réteg	GSM és 100BASE-TX
Adatkapcsolati réteg	GPRS és Ethernet
Hálózati réteg	Internet Protokoll (IP)
Átviteli réteg	TCP vagy UDP
Viszonylati réteg	TCP socket vagy szoftver
Megjelenési réteg	Rekordstruktúra vagy XML
Alkalmazási réteg	SQL szerver

5. MEGBÍZHATÓSÁG NÖVELÉSE

A járműkövető rendszereknek megbízhatóság szempontjából két fő tulajdonságuk van:

- adatátvitel biztonsága,
- on-line rendelkezésre állás

Az első tulajdonságon az értendő, hogy a járműről gyűjtött adatok változatlan tartalommal bekerülnek az adatbázisba. Ehhez meg kell határozni egy maximális időtartamot, ami után az adatot elveszettnek tekintjük.

A második tulajdonság azt fejezi ki, hogy a járműkövető rendszer működési ideje alatt, egy adott járműfedélzeti egység mekkora időtartamban van on-line kapcsolatban a kommunikációs szerverrel.

5.1 Járműfedélzeti egység biztonsága

Az adatátvitel biztonságának növeléséhez a rendszer minden elemét meg kell vizsgálni. A járműfedélzeti egységnek robusztus, EMC védett, járműipari kivitelű eszköznek kell lennie. Az egységnek képesnek kell lennie off-line működésre, azaz a kommunikációs kapcsolat megszakadása estén is gyűjtenie és tárolnia kell az adatokat. Fontos, hogy legyen saját – a jármű elektromos rendszerétől független – háttérakkumulátora.

5.2 A hálózati kapcsolat biztonsága

A nyilvános GSM hálózat biztonságát nem lehet befolyásolni. A hatósági előírások alapján két olyan minőségi paraméter nyilvános, ami az adatátvitel biztonságát befolyásolja. Az első a bithiba arány (hibásan átvitt bitek száma/ összes átvitt bitek száma), azonban ez IP alapú hálózatok esetén nem értelmezhető, ezért 0-nak tekinthető. A GSM hálózatok felsőbb rétegeiben még értelmezhető a bithiba, azonban az automatikus hibajavító eljárásoknak köszönhetően, IP szinten (elméletileg) a fizikai réteg hibái csak az adatátviteli sebesség csökkenésében jelentkeznek. Meg kell jegyezni azonban, hogy a GPRS szolgáltatás esetleges adatátviteli hibáiért a GSM szolgáltatók nem vállalnak felelősséget. A másik fontos minőségi jellemző a csomagvesztés (elveszett csomagok száma / feladott csomagok száma), amelyre a szolgáltató szintén nem vállal felelősséget, mivel a két végpont között előfordulhatnak az ő felelősségi területén kívül eső hálózati eszközök. Ez a hiba mindazonáltal csak az UDP használata esetén áll fenn, a TCP újraküldési algoritmusa kiküszöböli ezt a problémát.

További veszélyeket rejt a nyilvános internet használata. Ebben az esetben a szerver és a fedélzeti egységek is bármely internet használó számára elérhetők, így különböző rosszindulatú támadásnak vannak kitéve (pl.: túlterheléses támadás, adatforgalom eltérítése stb.).

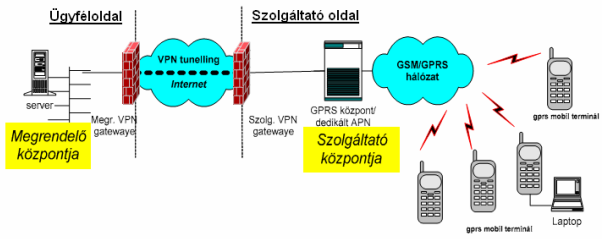
5.3 A kommunikációs szerver és az adatbázis biztonsága

A kommunikációs szerver esetén adatvesztés abban az esetben fordulhat elő, amennyiben egy nem várt hiba fellépése során a program üzemszerű futása megszakad (pl.: végtelen ciklusba fut, egy változó túlcsordul stb.), és így a memóriában lévő adatok elvesznek. Amennyiben a kommunikációs szerver és az adatbázis nem egy hardveren fut, úgy a helyi hálózati kommunikáció is okozhat hibákat, aminek kiküszöbölése szintén a kommunikációs szerver feladata.

5.4 Módszerek a biztonság növelésére

A járműfedélzeti egységgel kapcsolatos minimális követelményeket az 5.1-es fejezetben megadásra kerültek. Ebben a fejezetben - cikk céljának megfelelően - a hálózati kapcsolat, és a szerver biztonságával foglalkozunk.

Mint a fentiekben bemutattuk a GSM hálózat minőségi paraméterei (QoS), valamint az adatátviteli út felépítse a hálózati rétegig adottak. Azonban a GSM szolgáltatók biztosítanak egy olyan lehetőséget, amivel az adatátvitel a biztonsága nagyban növelhető. Egyéni mobilinternet szolgáltatás esetén minden ügyfél ugyanazt az GPRS elérési pontot (APN) használja. Így az ügyfelek bármely IP címmel rendelkező gépet elérnek és elérhető is válnak bárki számára. Ennek megfelelően a szerver fogadó portjának is elérhetőnek kell lennie minden IP címről. Szerver oldalon tűzfal segítségével le lehetne korlátozni a bejövő kapcsolatokat, azonban alapszolgáltatás használata esetén a mobil eszközök dinamikus IP címmel rendelkeznek, azaz minden hálózati bejelentkezésnél más IP címet kapnak. Magasabb szintű üzleti előfizetésnél lehetőség van a 4. ábrán látható adatátviteli út kiépítésére.



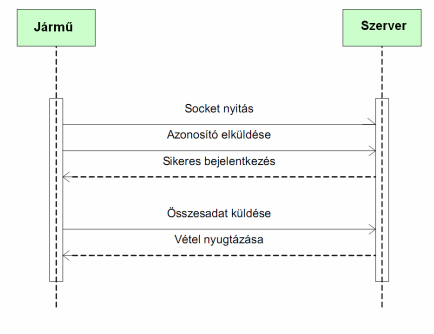
4. ábra: Biztonságos hálózati kapcsolat kialakítása (forrás: T-Mobile)

Ilyenkor a mobil eszközök (járműfedélzeti egységek) egy dedikált APN-hez kapcsolódnak, amelyet csak az adott flotta SIM kártyáival lehet elérni. A szolgáltató az ügyféllel egy erős titkosítással ellátott VPN (Virtual Private Network) csatornát alakít ki. A járműfedélzeti eszközök ezen keresztül érik el a kommunikációs szervert. Ebben a struktúrában a fedélzeti egységek csak a kommunikációs szerver látják, amely tűzfal mögött van, így védett a külső támadásokkal szemben. Megállapítható, hogy ez a rendszer nagyfokú biztonságot garantál mind a járműfedélzeti egységek, mind pedig a kommunikációs szerver számára.

Az adatátviteli út kialakításánál a protokoll definiálása a következő lépés. Meg kell határozni a szállítási, viszonylati és megjelenési réteget. A 4. fejezet alapján a megbízható kommunikációt igénylő alkalmazások számára a TCP a megfelelőbb, ez biztosítja a szállítási és viszonylati réteget. Erre kell építeni egy felsőbb szintű protokollt, amely biztosítja a kommunikációt a szerver és a járművek között. Erre a feladatra a – már bemutatásra került – XML-t választottuk, mivel sok előnye mellett, az egyetlen jelentősebb hátránya (nagyobb mennyiségű adatot kell átvinni) manapság már sem többletköltséget, sem pedig jelentős sebességcsökkenést nem okoz. A protokollal szemben a következő fontosabb követelményeket támasztottuk:

- A járműfedélzeti egység kezdeményezi a kapcsolat felépítését. Amint az eddig használt terminológia is utalt rá, a járművek a kliensek, a központ pedig a szerver.
- A kapcsolat állandó, annak lebontását, csak hiba esetén lehet kezdeményezni.
- Ha nincsen kapcsolat, akkor létre kell hozni, amint lehetséges.
- A protokollon az adatcsomagokat XML 1.1-es szabványnak megfelelő formátumban kell átvinni.
- Az adatokat előre definiált XSD alapján ellenőrizni kell. Csak megfelelő adatcsomagokat lehet elfogadni.
- Az elküldött csomagokat egyedi azonosítóval kell ellátni.
- Minden elküldött csomagra nyugtával kell válaszolnia a fogadónak. A nyugta lehet pozitív, vagy negatív.
- Addig kell ismételni egy adat küldését, amíg valamilyen nyugta nem érkezik.
- A nyugtára 30 másodpercet kell várni. Ha nem érkezett meg ezen időn belül, bontani kell a kapcsolatot. Ezzel a módszerrel a „beragadt” TCP socketek okozta hibák kiküszöbölhetőek.
- Le kell kezelni a többszörös nyugta és adat érkezését. Ehhez az egyedi csomagazonosítót kell használni.
- A csomagazonosító egy 16 bites előjelnélküli szám, mely minden egyes elküldött csomagnál növekszik. A csomagazonosító újraindulhat újracsatlakozás, valamint a 16 bites maximum érték (65535) elérése esetén.
- Ellenőrizni kell a csomagazonosító folytonosságát, hogy felderíthetők legyenek a kimaradt csomagok.
- A protokoll változásorientált. Csak azok az adatok kerülnek továbbításra, melyek megváltoztak az előzőleg elküldött csomag óta. Ez a módszer jelentősen csökkenti az átvitt adatmennyiséget.
- Minden járműről érkező adatcsomagnak tartalmaznia kell az alábbi adatokat:
    - Csomagazonosító
    - Idő másodperc pontossággal
    - GPS adatok (státusz, hosszúság, szélesség, irányszög sebesség)
- A kapcsolat felépülésekor a járműfedélzeti berendezésnek be kell jelentkezni szerverre, az egyedi azonosítója elküldésével.
- A járműfedélzeti egység addig nem küldhet adatokat, amíg nem sikerült bejelentkeznie. Az ilyen adatcsomagokat a szerver eldobja.

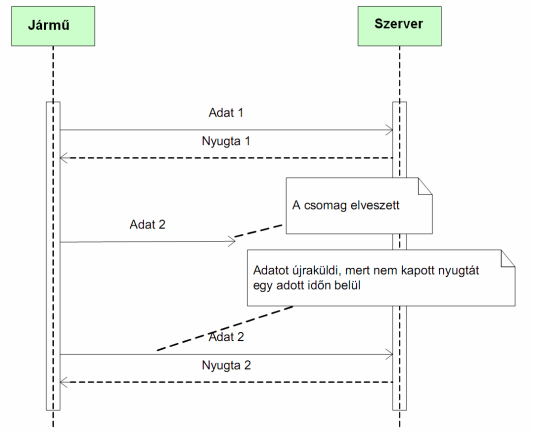
A kapcsolat létrejötte után a jármű elküldi egyedi azonosítóját (5. ábra), amire a szerver egy nyugtával válaszol.



5. ábra: A bejelentkezés folyamata

A „handshake” sikeres elbonyolítás után a jármű az összes mért adatát elküldi, ezzel biztosítván a kiindulási értékeket a változáskövetéshez.

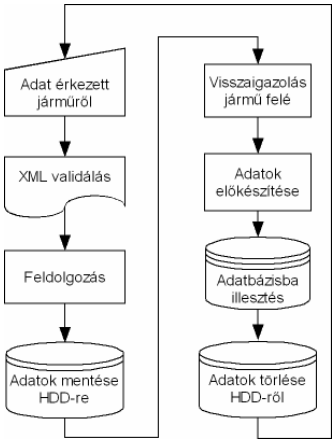
Ezután a ciklikus adatküldés (6. ábra) elindul. A jármű oldaláról ez esemény- és idővezérelt.



6. ábra: Az adatküldés folyamata

Normál esetben a jármű periodikusan küldi a mért adatokat, azonban bizonyos – előre definiált események (pl.: hirtelen üzemanyagszint csökkenés) – azonnali üzenetküldést generálnak.

A kommunikációs szerver esetén a biztonsági szempontból a legfontosabb a már beérkezett és visszaigazolt adatok megfelelő kezelése. A jármű felé visszaigazolt adatok a szerver felelősségi körébe tartoznak, azok a fedélzeti egységről törlődnek. Ennek megfelelően a beérkezett adatokat a validálás és előfeldolgozás után a helyi merevlemezen (HDD) kell tárolni, majd ezt követően a nyugtát a járműnek megküldeni (7. ábra). Ezután el lehet végezni a teljes feldolgozást és az adatbázisba illesztést.



7. ábra: A biztonsági adatkezelés folyamata

Ennél a módszernél az adatok egészen addig a járműfedélzeti egység háttértárolóján elérhetők, amíg a szerver merevlemezén mentésre nem kerültek. Így programhiba esetén nem történik adatvesztés, továbbá az adatbázis-szerver leállása esetén sem vesznek el az adatok. Helyes működéshez elengedhetetlen, hogy a kommunikációs szerver csak akkor törölje a helyben tárolt adatokat, ha pozitív visszajelzést kapott az adatbázisszervertől az beillesztési parancs (INSERT) végrehajtásáról. A tárolás biztonságának növelése érdekében a kommunikációt és az adatbázist kiszolgáló hardvereknek nagy megbízhatóságú szerverarchitektúrát kell választani (redundáns merevlemez, redundáns adatbázis stb.).

5.5 Az on-line rendelkezésre állás növelése

Az on-line rendelkezésre állás legjelentősebb befolyásoló tényezője a GSM hálózat. Ezt befolyásolni nem lehet, azonban érdemes megvizsgálni a korlátait. Az első paraméter a területi lefedettség, különös tekintettel a GPRS lefedettségre. A Nemzeti Hírközlési Hatóság adatai szerint mindhárom szolgáltató legalább 99%-os területi lefedettséggel rendelkezik, és elméletileg a teljes szolgáltatási területén elérhető a GPRS. A gyakorlatban ez némileg kevesebb, a GPRS területi lefedettség 97% körül alakul. Közúti járműkövető rendszereknél az arány még jobb is lehet, mivel a közutak lefedettségére nagyobb hangsúlyt helyeztek a hálózatok kiépítésénél. A tapasztalatok azt mutatják, hogy a vasútvonalak mentén roszszabb a helyzet.

A következő fontos paraméter a rendelkezésre állás. Az éves rendelkezésre állást hatóságilag előírt módon kell meghatározni a következő módon:

$$\text{Re}\,ndelkezésrállás = \frac{AUT}{AUT + ADT} \times 100\% \quad (1)$$

Ahol:

ADT a teljes kiesési idő (a szolgáltatás megszűnése és annak visszaállítása között eltelt idő),

AUT a működési idő (a szolgáltatás elindítása és annak megszűnése között eltelt idő),

AUT+ADT a teljes megfigyelési idő.

A rendelkezésre állás meghatározásakor nem kell figyelembe venni az előfizetői végberendezéseket, a rádiós területi fedettség veszteségeit, és a forgalmi torlódásokból eredő veszteségeket. A rendelkezésre állás számításakor az egyes bázisállomások teljes kiesését kell figyelembe venni. Az előírások szerint a GSM PLMN (Public Land Mobile Network) rádiótelefon rendszer bázisállomásainak az előre tervezett szolgáltatási szünetek kivételével (< 4 óra/év) az év minden napján napi 24 órán keresztül éves átlagban 95%-os rendelkezésre állással a rendeltetésszerű használatra alkalmasnak kell lennie. Ugyanez az érték vonatkozik az internet szolgáltatásra is, azonban a tavalyi adatok alapján a szolgáltatók ennél jobbat, 99,61 %-ot értek el. Fontos felhívni a figyelmet a torlódásokból eredő veszteségekre, mivel a GPRS kapcsolat alacsonyabb prioritású mint a hanghívás. Ezért terhelt cellákban előfordulhat, hogy a hálózati torlódások miatt bizonyos ideig nem lehet az adatokat eljuttatni a járműről a központba.

A fenti értékek alapján elmondható, hogy a nyilvános GSM hálózat általános célú járműkövetésre kiválóan alkalmas, azonban biztonsági igényű alkalmazások (pl.: vasúti területen) esetén nem megfelelő. Vasúti területen azonban érdemes számba venni – a néhány éven belül kiépítésre kerülő – GSM-R (GSM for Railway) rendszert.

Az on-line rendelkezésre állás megfelelő szinten tartásához elengedhetetlen a megbízható szoftverek kifejlesztése a járműfedélzeti egységben és a szerver oldalon.

A szerverszoftver biztonságáról már esett szó az előzőekben, és láttuk, hogy megfelelő adatkezeléssel biztosítani lehet, hogy veszteségmentes legyen az adatküldés. Ennél nagyobb problémát jelent a rendelkezésre állás biztosítása, mivel ebben az esetben a szoftver illetve hardver leállásokat (melyek az adatbiztonságot nem befolyásolják) a lehető legnagyobb mértékben ki kell küszöbölni. A hardver esetében a fent említett nagy megbízhatóságú szerverarchitektúrát választva, és azt megfelelő környezetben (temperált szerverszoba, szünetmentes tápellátás, redundáns internet kapcsolat) elhelyezve, a rendelkezésre állás magas szinten tartható. A szoftver megbízhatóságát a következő szempontok figyelembe vételével lehet növelni:

- pontos specifikáció,
- megfelelően lehatárolt funkciók,
- moduláris felépítés,
- egyszerűségre, átláthatóságra és tesztelhetőségre való törekvés,
- részletesen megtervezett tesztek végrehajtása (modulok verifikációja, teljes szoftver validációja)
- funkcionális, terheléses, biztonsági, hibakezelési és tartós tesztek,
- folyamatos dokumentálás.

Mindezen kritériumok megvalósításának részletezése meghaladja a cikk kereteit, azonban az ennyiből is látszik, hogy a magas rendelkezésre állású szoftverek megvalósítása, nagy körültekintést igénylő, bonyolult és sokrétű feladat.

6. ÖSSZEGZÉS

A cikkben bemutatatásra került a járműkövető rendszerek adatátviteli technológiáinak ismertetése. Az OSI-modell alapján ismertettük az egyes rétegek esetében használható szabványokat és protokollokat. Kiválasztottuk a megbízhatóság szempontjából legoptimálisabb megoldást. Továbbá kifejlesztettünk egy – a megjelenési rétegben használható – nagy biztonságot nyújtó kommunikációs protokollt járműkövető rendszerek számára. Megvizsgáltuk a szerver oldali adatkezelés biztonsági kérdéseit, és bemutattunk egy – a kifejlesztett protokollal együtt használható – biztonságos adatkezelési módszert. Végül megvizsgáltuk az on-line rendelkezésre állást befolyásoló tényezőket a GSM hálózat és a szoftverfejlesztés szempontjából. Megállapítottuk, hogy biztonságkritikus alkalmazások esetén a GSM hálózat nem ad kielégítő rendelkezésre állást, különösen tekintettel a torlódásokból eredő veszteségekre. Ezért vasúti területen mindenképpen számba kell venni a jövőben a GSM-R használatát.

HIVATKOZÁSOK

Aradi Sz. (2007) Vasúti távfelügyeleti rendszer mozdony fedélzeti berendezés alkalmazásával; *Vezetékek Világa 2007/1. pp. 27-8.*

Aradi Sz. (2007) Server Architecture Development for On-line Tracking of Large-sized Vehicle Fleet; *Periodica Polytechnica*

ISO 7498:1984 Open Systems Interconnection - Basic Reference Model; *http://standards.iso.org/ittf/PubliclyAvailableStandards*

IP, TCP, UDP specifikáció; *http://tools.ietf.org*

Hírközlési és Informatikai Tudományos Egyesület: Távközlő hálózatok és informatikai szolgáltatások; *Online könyv, http://www.hte.hu*

Hírközlés-statisztikai Adatbázis; *Nemzeti Hírközlési Hatóság, http://www.nhh.hu/hirk_stat*

Interneten keresztül történő titkosított csatlakozás rendszertechnikai rajza; *T-Mobile, http://t-mobile.hu*

R. Lyu, M. (1996) Handbook of Software Reliability Engineering; *McGraw-Hill publishing, 1995, ISBN 0-07-039400-8*

Extensible Markup Language (XML) 1.1 (Second Edition); http://www.w3.org/TR/xml11

XML Schema; http://www.w3.org/XML/Schema