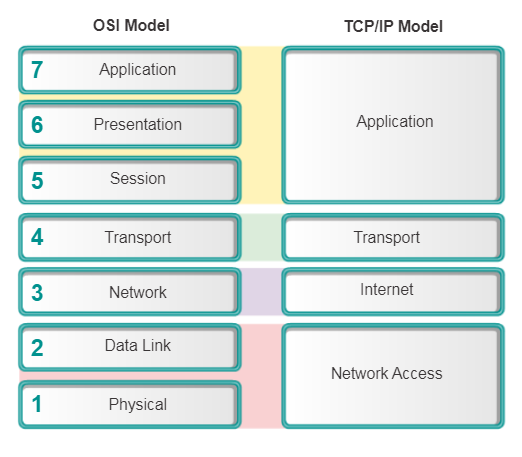
**ALKALMAZÁSI, VISZONY ÉS MEGJELENÍTÉSI réteg**

A hálózati szakemberek mind szóban, mind pedig az írott műszaki dokumentációban az OSI és a TCP/IP modelleket egyaránt alkalmazzák (lásd ábra). Ezeket a modelleket használhatják a protokollok és az alkalmazások viselkedésének leírására.

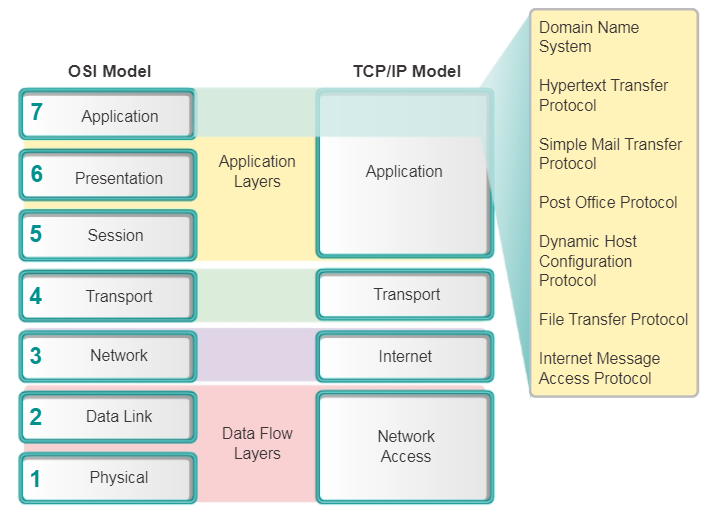
Az OSI-modellben az adat rétegről-rétegre halad, elindulva a forrásállomás alkalmazási rétegéből, haladva lefelé a hierarchiában a fizikai rétegig, át a kommunikációs csatornán egészen a célállomásig, ahol aztán elindul felfelé a hierarchiában egészen az alkalmazási rétegig.

**Az alkalmazási réteg mind az OSI mind a TCP/IP-modellnek a legfelső rétege**. A TCP/IP-modell **alkalmazási rétege számos, a végfelhasználói alkalmazások szolgáltatásaihoz szükséges protokollt tartalmaz**. A TCP/IP alkalmazási rétegbeli protokolljainak funkciói nagyjából az OSI-modell felső három rétegének, az alkalmazási-, megjelenítési- és viszonyrétegnek felelnek meg. **Az alkalmazásfejlesztők és gyártók az OSI-modell 5., 6. és 7. rétegére hivatkoznak hálózati hozzáférést igénylő termékeik, pl. egy web böngésző kapcsán**.



**Az alkalmazási réteg**

**A végfelhasználóhoz az alkalmazási réteg van a legközelebb**. Ahogy azt az ábra is mutatja, **ez az a réteg, amely az interfészt biztosítja az általunk kommunikációra használt alkalmazások és a mögöttes hálózat között, amelyen üzeneteink továbbítódnak**. Az alkalmazási rétegbeli **protokollokat a forrás- és célállomásokon futó programok közötti adatcserére használjuk**. Számos alkalmazási rétegbeli protokoll létezik, és mindig vannak új, fejlesztés alatt álló protokollok is. **A legismertebb alkalmazási rétegbeli protokollok a HTTP (Hypertext Transfer Protocol), az FTP (File Transfer Protocol), a TFTP (Trivial File Transfer Protocol), az IMAP (Internet Message Access Protocol) és a DNS (Domain Name System) protokoll.**



**A megjelenítési réteg**

A megjelenítési rétegnek három fő funkciója van:

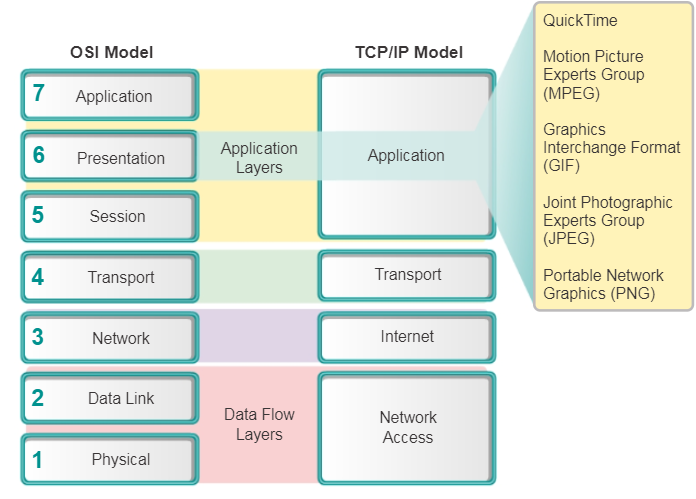
* **Megjeleníti a forrásállomásról származó adatokat, vagy átalakítja azokat a célállomás által igényelt formára.**
* **Tömöríti az adatokat egy a célállomás által kitömöríthető formátumban.**
* **Titkosítja az adatokat az átvitelhez, valamint a célállomáson visszafejti azokat**.

Amint az ábrán látható, a **megjelenítési réteg az alkalmazási réteg adatait alakítja át, valamint határozza meg az egyes fájlformátumokra vonatkozó szabványokat**. Az ismertebb videó szabványok közé tartozik a **QuickTime és az MPEG** (Motion Picture Experts Group). A QuickTime egy videóra és hangra vonatkozó Apple Computer által jegyzett szabvány, míg az MPEG szintén egy videó- és hangtömörítésre, valamint kódolásra szolgáló formátum.

A **hálózatokon használt ismertebb grafikus képformátumok közé tartozik a GIF** (Graphics Interchange Format), a **JPEG** (Joint Photographic Experts Group) és **a PNG** (Portable Network Graphics). A GIF és a JPEG rasztergrafikus képek tömörítésére és kódolására szolgáló szabványok. A PNG-t a GIF formátum bizonyos korlátainak kiküszöbölésére, majd végül a formátum lecserélésére tervezték.

**A viszonyréteg**

Ahogy a neve is utal rá, a viszonyréteg **feladatai a forrás- és célalkalmazások közötti párbeszédek (munkamenetek) létrehozása és fenntartása**. **A viszonyréteg kezeli a párbeszédek kialakításához, fenntartásához, valamint a megszakadó vagy hosszabb ideje tétlen viszonyok újraindításához szükséges információcserét.**

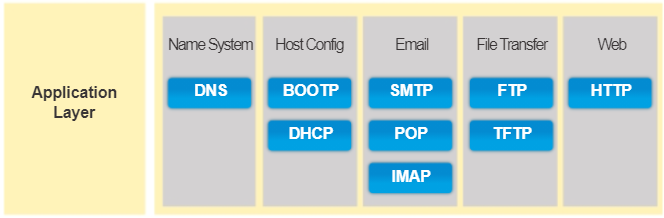


*Míg az OSI-modell megkülönbözteti az alkalmazási-, megjelenítési- és viszony funkciókat, addig a széles körben ismert és használt TCP/IP-alapú alkalmazások egyesítik a három réteg funkcionalitását.*

A TCP/IP alkalmazási protokolljai számos népszerű internetes kommunikációs szolgáltatás formátumát és vezérlőinformációit határozzák meg. Ezen TCP/IP-protokollok közé tartoznak az alábbiak:

* **Tartománynév-kezelő rendszer (Domain Name System, DNS)** - Ez a protokoll az internetes nevek IP-címekhez történő hozzárendelését végzi.
* **Telnet** - Szerverekhez és hálózati eszközökhöz való távoli hozzáférésre használjuk.
* **Egyszerű levéltovábbító protokoll (Simple Mail Transfer Protocol, SMTP)** - Ez a protokoll e-mail üzeneteket és csatolmányokat továbbít.
* **Dinamikus állomáskonfiguráló protokoll (Dynamic Host Configuration Protocol, DHCP)** - A protokollt arra használjuk, hogy egy állomáshoz IP-címet, alhálózati maszkot, alapértelmezett átjárót és DNS szervert rendeljünk.
* **Hiperszöveg továbbító protokoll (Hypertext Transfer Protocol, HTTP)** - Ez a protokoll a világháló weboldalait felépítő fájlokat továbbítja.
* **Fájlátviteli protokoll (File Transfer Protocol, FTP)** - A protokollt rendszerek közötti interaktív fájlátvitelre használjuk.
* **Fájlátviteli protokoll (File Transfer Protocol, FTP)** - Ezt a protokollt kapcsolat nélküli aktív fájlátvitelre használjuk.
* **Rendszerbetöltő protokoll (Bootstrap Protocol, BOOTP)** - Ez a protokoll a DHCP protokoll előfutára. A BOOTP egy hálózati protokoll, amelyet a rendszerindítás során az IP-címzési információk megszerzésére használunk.
* **Postahivatal protokoll (Post Office Protocol, POP)** - A protokollt a levelezőkliensek az e-mailek távoli szerverről történő letöltésére használják.
* **Internetes levélhozzáférési protokoll (Internet Message Access Protocol, IMAP)** - Ez egy másik, szintén a levelek letöltését szolgáló protokoll.

A kommunikációs folyamat során az alkalmazási rétegbeli protokollokat mind a forrás-, mind pedig a célállomás egyaránt használja. Hogy a kommunikáció sikeres legyen, a forrás- és célállomásokon használt alkalmazási rétegbeli protokolloknak egymással kompatibilisnek kell lenniük.



Feladat ehhez kapcsolódik!!!!

## Az alkalmazási protokollok és a végfelhasználói alkalmazások együttműködése

Amikor egy hálózati eszközön - legyen az PC, laptop, tablet, okostelefon vagy bármilyen más hálózatra csatlakoztatott eszköz - hozzáférünk valamilyen információhoz, **akkor nem biztos, hogy az fizikailag is az adott eszközön van tárolva**. **Ebben az esetben az információ eléréséhez egy kérést kell küldeni az adatokat tároló eszköznek**. Az **egyenrangú** (P2P, peer-to-peer) **hálózati modellben az adatok elérése dedikált szerver használata nélkül, egy társ eszközről (peer) történi**k.

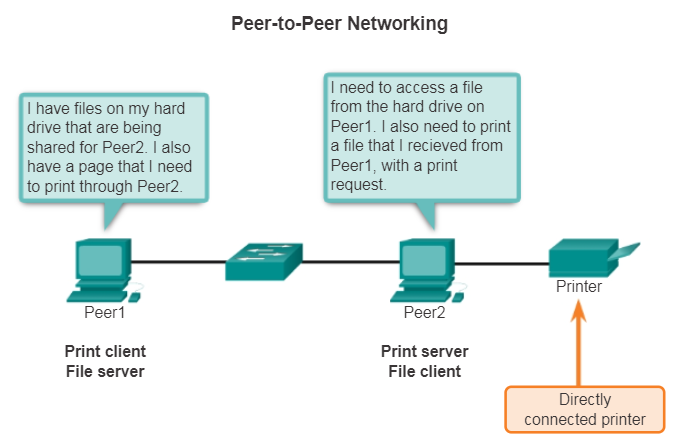
**A P2P hálózati modell két részre bontható: P2P-hálózatokra és P2P-alkalm**azásokra. Mindkét résznek hasonlók a tulajdonságaik, de a gyakorlatban egészen másként működnek.

**P2P-hálózatok**

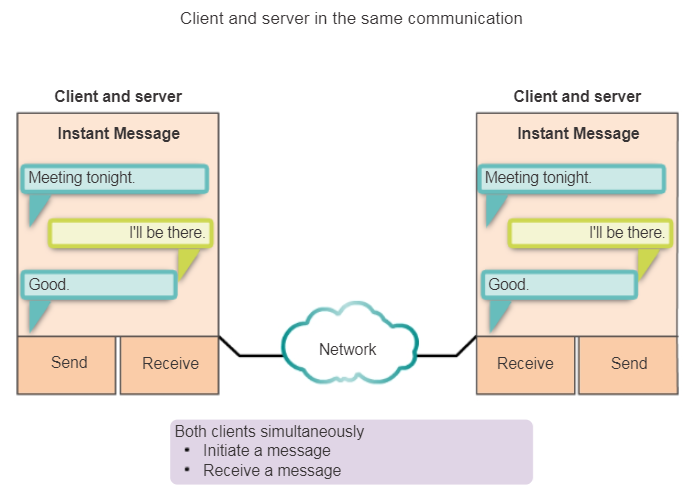
Egy **P2P-hálózatban két vagy több számítógép csatlakozik egymáshoz a hálózaton keresztül úgy, hogy dedikált szerver nélkül oszthatják meg egymás között az erőforrásaikat** (pl. nyomtatókat vagy fájlokat). **Minden csatlakoztatott végberendezés** (más néven peer) **működhet szerverként és kliensként is egyben**.

*Egy számítógép az egyik adatátviteli folyamatban betöltheti a szerver szerepkörét, míg egy másikban ezzel egyidejűleg kliens is lehet*. **A szerver és a kliens szerepköröket az egyes kérések határozzák meg.**

*Példa lehet erre egy egyszerű otthoni hálózat két számítógéppel, ahogy az az ábrán is látható. A példában Peer2-nek van egy hozzá USB-n közvetlenül csatlakoztatott nyomtatója és be van állítva a hálózati megosztása úgy, hogy azon Peer1 tudjon nyomtatni. Peer1 pedig egy meghajtó vagy egy mappa hálózati megosztására van beállítva. Ez Peer2-nek lehetővé teszi a megosztott mappához való hozzáférést és abba fájlok mentését. Egy ilyen hálózat a fájlok megosztásán kívül a felhasználóknak a hálózati játékok használatát, vagy egy internetkapcsolat megosztását is biztosíthatja.*

**A P2P-hálózatok decentralizálják a hálózat erőforrásait.** Az adatok dedikált szerverek helyett bárhol és bármely csatlakoztatott eszközön megoszthatók. A **legtöbb mai operációs rendszer további szerverszoftver igénye nélkül támogatja a fájl- és nyomtatómegosztást**. A P2P-hálózatok a jogosultságok kezelésére nem alkalmaznak központosított felhasználói fiókokat vagy hozzáférési szervereket. Ezért a néhány számítógépnél többet tartalmazó hálózatok esetében is már bonyolult lehet a biztonsági és a hozzáférési házirendek érvényre juttatása. **A felhasználói fiókokat és a hozzáférési jogosultságokat az egyes eszközön egyedileg kell beállítani**.

**Egy peer-to-peer (P2P) alkalmazás lehetővé teszi, hogy egy eszköz ugyanabban a kommunikációban egyszerre kliens és szerver is lehessen** (lásd ábra). Ebben a modellben minden kliens szerver is és minden szerver kliens is egyben. **Mindkettő kezdeményezhet kommunikációt és a kommunikációs folyamatban egyenrangúnak tekintendők**. **A P2P-alkalmazások megkövetelik** azonban, hogy **minden végberendezés biztosítson egy felhasználói felületet és egy háttérszolgáltatást is futtasson**. Amikor **elindítunk egy adott P2P-alkalmazást, az betölti a szükséges felhasználói felületet és a háttérszolgáltatásokat, majd ezt követően az eszközök már közvetlenül tudnak egymással kommunikálni.**



**Bizonyos P2P-alkalmazások úgynevezett hibrid rendszert használnak**, ahol az **erőforrások megosztása ugyan decentralizált**, de az **erőforrások helyeire mutató indexeket már egy központi címtárban tárolják.**

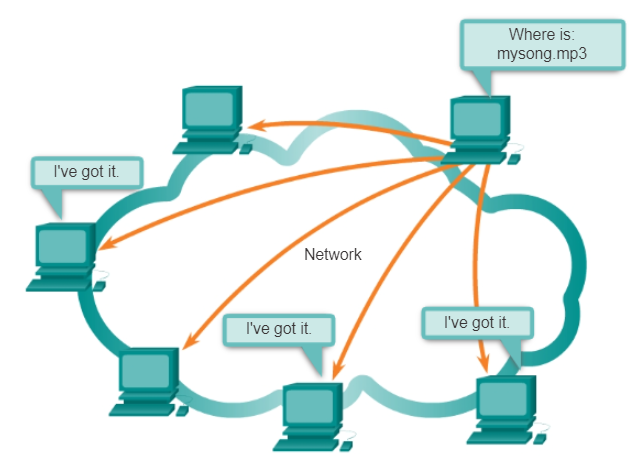
**A hibrid rendszerekben minden csomópont (peer) hozzáfér egy indexszerverhez, ahonnan lekérdezheti a más csomópontokon tárolt erőforrások helyét**. Az indexszerver segíthet a két csomópont összekapcsolásában, de azt követően a csomópontok közötti kommunikáció már a szervertől függetlenül zajlik.

**P2P-alkalmazásokat használhatunk P2P-hálózatokban, kliens-szerver hálózatokban és az interneten keresztül is**.

P2P-alkalmazást használva a hálózat valamennyi számítógépe kliensként és szerverként is szolgálhat a hálózat azon többi számítógépe számára, amelyek ugyanazt az alkalmazást futtatják**. Az ismertebb P2P-alkalmazások közé tartoznak a következők:**

* eDonkey
* eMule
* Shareaza
* BitTorrent
* Bitcoin
* LionShare

**Bizonyos P2P alkalmazások a Gnutella protokollra épülnek. A Gnutella lehetővé teszi felhasználóinak, hogy megosszák egymással a merevlemezeiken lévő fájlokat***. Amint az az ábrán is látható, egy Gnutella kompatibilis kliensszoftver lehetővé teszi a felhasználóknak, hogy az interneten Gnutella szolgáltatásokhoz kapcsolódjanak, és más Gnutella csomópontok által megosztott állományokat találjanak meg és érjenek el.* A Gnutella hálózat elérésére számos kliensalkalmazás létezik, közöttük a BearShare, a Gnucleus, a LimeWire, a Morpheus, a WinMX és a XoloX.



Míg az alapprotokollt a Gnutella Fejlesztői Fórum (Gnutella Developer Forum) gondozza, addig az alkalmazásforgalmazók gyakran fejlesztenek hozzá kiterjesztéseket, hogy a protokollt az alkalmazásaikba illeszthessék.

**Számos P2P alkalmazás nem használ központi adatbázist, valamennyi fájlt a peer-ek tartanak nyilván.** Lekérdezésre **a hálózat valamennyi eszköze elmondja a többieknek, hogy rajta keresztül mely fájlok érhetők el,** **majd a fájlmegosztó protokollt és a szolgáltatásokat használják az állományok megkeresésére.**

**A kliens-szerver modellben az információt kérő eszközt kliensnek, a kérésre válaszoló eszközt pedig szervernek vagy kiszolgálónak nevezzük**. **A kliens- és szerverfolyamatokat az alkalmazási réteghez soroljuk**. *A párbeszédet a kliens kezdeményezi azzal, hogy adatokat kér a szervertől, amely aztán a kliensnek egy vagy több adatfolyam elküldésével válaszol*. A kliensek és szerverek közötti kérések és válaszok formátumát az alkalmazási rétegbeli protokollok határozzák meg. **A tényleges adatátvitel mellett ez a párbeszéd a felhasználó hitelesítését, valamint az átvitt adatfájl azonosítását is megkövetelheti.**

*A kliens-szerver hálózat egy példája, amikor egy ISP e-mail szolgáltatását használjuk levelek küldésére, fogadására és tárolására.* Egy otthoni számítógépen lévő levelezőkliens egy kérést intéz az ISP levelezőszerveréhez egy olvasatlan levélért. A szerver ezután a válaszüzenetében elküldi a kért levelet a kliensnek.

Bár az adatok általában a szerverről áramlanak a kliens felé, bizonyos adatok mindig a kliens felől haladnak a szerver irányába. Az adatáramlás lehet mindkét irányban lehet egyenlő mértékű, de lehet akár nagyobb is a klienstől a szerver irányába. A kliens például tárolási célból átmásolhat egy fájlt a szerverre. Ahogy az ábra is mutatja, **a kliensről a szerverre történő adatátvitelt feltöltésnek (upload), míg a szerverről a kliensre történőt letöltésnek (download) nevezzük.**

