

## XOOPS

Készítsünk közösségi weblapokat cikksorral, megjegyzéssel, felhasználói blogokkal és egyébekkel felszerelve.

**A**z elmúlt néhány hónapban egy olyan tartalomkezelő programmal (CMS) ismerkedhettünk meg, amely megkönnyíti az utóbbi időben roppant népszerű blogok készítését. Hogy mi is pontosan a blog, arra jelenleg nem létezik egyértelmű és szigorú meghatározás, de a legtöbb ember valószínűleg egyetért abban, hogy egy olyan formáról van szó, melyben rövid cikkekből áll a tartalom, ahol a legfrissebb jelenik meg legelőször.

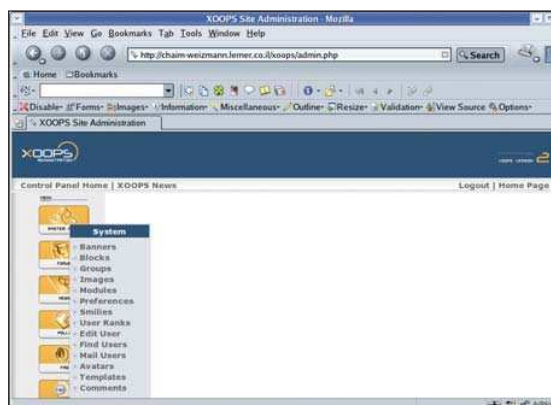
Létezik azonban egy nagyobb, kicsit eltérő típus is, amit jobb híján talán Slashdot-stílusnak nevezhetnénk. Ezeket a blogokat ténylegesen kis csoportok szerkesztik, ám van e mellett egy rengeteg tagból álló közösség is, amely részt vesz a napi munkában. Utóbbiak sok szempontból talán közelebb állnak a fórumokhoz és a hirdetőtáblákhoz mint a blogokhoz, de van elég sok naplószerű jellemzőjük is. Miután ugyanis a lap látogatója elolvasta a vezérfonalat adó cikket, elkészítheti saját weblogját, valamint elolvashatja a többi felhasználó bejegyzéseit.

Néhány csomag – közülük is elsősorban a Slash, amely a Slashdot alapjául szolgál, vagy a use.perl.org – kifejezetten a nyilvános és privát weblogok alapötlete köré épült, de más csomagok is felnöttek a feladathoz. Az egyik ilyen csomag az XOOPS, mely név valószínűleg az Extended Object Oriented Portal System (Kiterjesztett objektum-központú portál rendszer) rövidítéséből származik. Az XOOPS a PHPNuke és PostNuke családba tartozó rendszerek oldalhajtása.

Ebben a hónapban az XOOPS rendszerrel ismerkedünk meg, különös figyelmet szentelve a WeBLog (kiejtve: VÍ-bílog) modulnak, amelyet nyilvános és privát blogok létrehozására és felügyeletére terveztek. Vannak ugyan fenntartásaim az XOOPS-al kapcsolatban, bizonyos közösségek számára azonban kiváló eszköz lehet. Különösen két területen nehéz jobbat találni: rendkívül egyszerű telepíteni és karbantartani, továbbá minden felhasználó saját weblogot kaphat.

### Történelem

Az XOOPS kiváló példája a GNU General Public Licence hatékonyságának, a nyílt fejlesztési közösségnek, valamint annak, hogy nem feltétlenül káros minden szétválás a nyílt forrású fejlesztésekben. Az XOOPS története a PHP nyelven és MySQL adatbázis háttérrel készülő PHPNuke prog-



1. ábra Az XOOPS honlapgazda oldala, ahol kiválaszthatjuk a lapunkon megjelenő modulokat

rammal kezdődött. A PHPNuke akkoriban a PHP nyelven készülő, ThatWare nevezetű Slash klónon alapult, de utóbbinak már felhagytak a fejlesztésével.

Ami ez után történt, annak megítélése – finoman szólva – elég ellentmondásos, és jól példázza, hogy mi történhet akkor, amikor egy nyílt forrású fejlesztés eldurvul:

a PHPNuke számos fejlesztője úgy döntött, elszakadnak a projektől és saját fejlesztésüket PostNuke-nak nevezték el. Manapság a PHPNuke és a PostNuke egyaránt létezik, amivel a felhasználói és fejlesztői közösségek közti kapcsolat szenvedett némi csorbát.

Nos, úgy tűnik, hogy a XOOPS a PHPNuke objektum-központú átírata. Az XOOPS fejlesztők úgy gondolják, hogy a rendszert objektum alapon újraírva megnövekszik a biztonság, és könnyebbé válik a fejlesztés és a karbantartás. A jó hír az, hogy a fent felsorolt rendszerek mindegyike elérhető a GNU General Public License alatt.

Az XOOPS-ot szemmel láthatólag kisebb fejlesztőcsapat készíti, mint a PHPNuke-ot vagy PostNuke-ot. Mégis, a beépített és mások által készített modulok száma igen magas, valamint objektum-központú lehetőségeinek köszönhetően nagyon könnyen készíthetünk saját modulokat. Tulajdonképpen a modulok kis száma ellenére lenyűgöző, mennyire egyszerűen telepíthetőek. Ez azt jelenti, hogy inkább

XOOPS-t használjunk a többi Nuke-szerűség helyett? Nos, ez legalább annyira függ a meggyőződéstől és a tervezett felhasználástól, mint magától a technológiától, bár önmagában a tény, hogy az XOOPS objektumokat használ, tapasztalataim szerint az élvonalba emeli, legalábbis elméletben.

### Letöltés és telepítés

Miután áttekintettük a XOOPS és rokonainak őstörténetét, állítsuk be saját XOOPS kiszolgálónkat. Az XOOPS futtatásához szükségünk lesz egy MySQL kiszolgálóra – személy szerint a köztudottan elavult 3.23.58 verziót használom – valamint egy Apache kiszolgálóra (1-es vagy 2-es változat) telepített PHP támogatással.

Az XOOPS hosszú ideje az egyik legfigyelemreméltóbb és legegyszerűbben telepíthető program amivel találkoztam. A kódot az XOOPS weblapról töltöttem le (lásd a hálózati források szakaszt) majd az ideiglenes könyvtárban megnyitottam a *.tar.gz* állományt. A teljes HTML alkönyvtárat felmásoltam az Apache tartalom gyökerébe egy könyvtárba: */usr/local/apache/html/xoops*. Ezt követően az XOOPS telepítőt a */xoops* URL alól érhettem el.

A teljesen web-alapú telepítő lassan de biztosan vezet minket végig a folyamaton. Talán a leginkább elbizonytalanító rész az a lap, ahol az új adatbázist és táblákat létrehozunk és a jogosult MySQL felhasználó nevét és jelszavát kell megadnunk. Ezen kívül a telepítő lehetne kicsit szöszátyárabb a hibaüzenetek tekintetében, tekintve, hogy nem mindenki tapasztalt web/adatbázis programozó.

Azt kell mondjam, a telepítés nagyon egyszerű és gördülékeny volt. Visszatértem a rendszerem */xoops* URL-jére és máris egy egyszerű bemutatkozó képernyővel találkoztam, amely belépésre invitált. Beléptem a lap adminisztrátorként, és bár a főlap elég üres volt, gyorsan tovább tudtam lépni a képernyő bal felső részén található rendszergazdai menüre (1. ábra).

Az XOOPS kezelése eléggé magától értetődő, ha rájövünk, hogy XOOPS alatt minden modulnak számít. A legtöbb XOOPS gazda első dolga tehát további modulok feltelepítése. Lépjünk a kezelői (administration) menübe és válasszuk a Modules pontot. A képernyő tetején találjuk a már feltelepített modulokat, míg a lap alján sorakoznak azok, amelyek elérhetők de nincsenek telepítve.

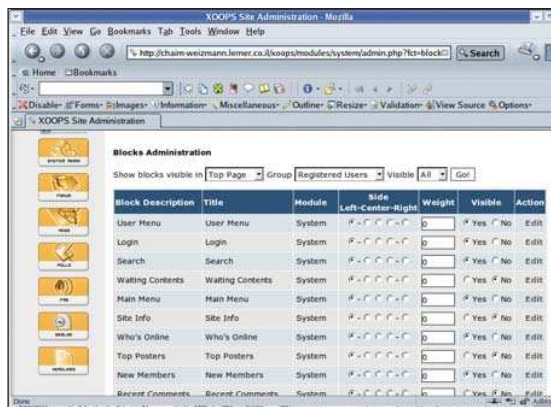
Utóbbiak telepítéséhez kattintsunk a modul neve melletti ikonra. Választásunk jóváhagyását követően megtekinthetjük milyen táblák jöttek létre a MySQL adatbázisunkban, milyen új elérési jogosultságok kerültek a rendszerbe valamint azt is, hogy a modul telepítése sikeres volt-e.

Tegyük fel, hogy így történt, és visszatérhetünk a Module Administration lapra, ahol láthatjuk, hogy a frissen feltelepített modul a képernyő aljáról átkerült a tetejére.

Az XOOPS oldal alapértelmezetten gyakorlatilag üres.

Ha egyéb elemeket szeretnénk a honlapunkhoz adni, akkor át kell lépniük a blokkok (Blocks) karbantartói lapjára.

A News (hírek) modul három blokkot tartalmaz – *bigstory*, *topics* és *top*. A fő blokk-karbantartó menüben a lapgazda beállíthatja mely blokkok legyenek láthatóak. A dolog igazi erejét azonban az egyes blokkokhoz tartozó szerkesztés (Edit) menüben találjuk, ahol megadhatjuk, hogy a képernyőn hol jelenjen meg az adott blokk – oldalt, középen vagy éppen teljesen máshol.



2. ábra Készítsünk egyedi lapkinézetet a blokkok helyzetének megadásával a Blocks Administration menüben

### Modulok

Minden XOOPS modul jól meghatározott és egységes könyvtárakba rendezett HTML, PHP és SQL részekből áll. Az osztálydefiniciókat mindig a *class* könyvtárban, az adatbázis definiciókat az *SQL*-, a blokkokat a *blocks*, míg a képeket az *images* könyvtárban találjuk. A beállítás és verzió információk mindig a *xoops\_version.php* állományban, a modul gyökerében lelhetők fel.

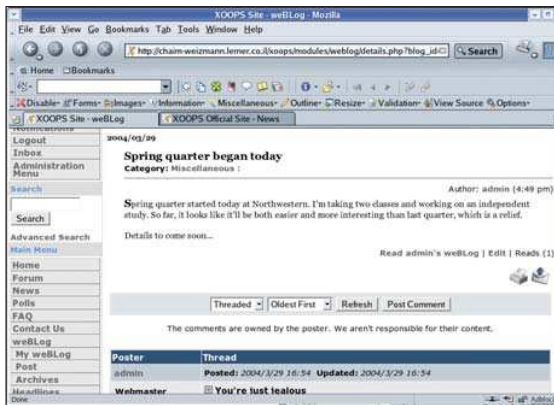
Az új modul telepítése mindössze annyiból áll, hogy letöltés után a könyvtárát áthelyezzük *modules* könyvtárba. Tehát ha letöltöttük a *foo* nevű modult, és a XOOPS könyvtárunk a */usr/local/apache/html/xoops*, egyszerűen csak másoljuk a *foo* könyvtárat a */usr/local/apache/html/xoops/modules* helyre. Legközelebb, amikor a Module Administration lapra lépünk, a modult már ott találjuk, telepítésre és beállításra készen.

Az egyik fontos dolog amit a XOOPS fejlesztők szem előtt tartottak, az adatbázis továbbfejlesztése. A XOOPS modulok PHP és HTML részeinek fejlesztési irányát nem nehéz elképzelni. Egy PHP-t és SQL-t egyaránt alkalmazó modul frissítése ugyan már kicsit keményebb dió lehet, de úgy néz ki, a fejlesztők ezt sikerrel megoldották, és tiszteletre méltó munkát végeztek. A XOOPS modul SQL könyvtárában nem csak a MySQL definiciókat tartalmazó *mysql.sql* állományt találhatjuk meg, hanem az ALTER TABLE parancsokat tartalmazó frissítő állományt is, amely a régi verzióról az újra változtatja adatbázisunkat. Mivel a modulok táblái a rendszer többi részétől elszigetelve léteznek, a többitől teljesen függetlenül, és szinte bármilyen sorrendben frissíthetjük az XOOPS moduljait.

### WeBlog

Miután ráérettünk a modulokkal való munkára, hozzáláthatunk *Hiroyuki Sakai* WeBlog moduljának telepítéséhez. Most végre olyan jellemzőkkel láthatjuk el programunkat, amelyeket eredetileg kerestünk: egy közösség által is használható webnapló rendszert építünk.

A WeBlog telepítéséhez töltsük le az állományt a főoldalról (lásd a forrásokat). Akárcsak a legtöbb XOOPS modul esetében, most is ki kell csomagolnunk a *tar\_zxf* parancssal, majd a legfelső szintű weblog könyvtárat átmásolnunk



3. ábra Siker! A beállított lapon megfigyelhetjük a felhasználók bejegyzéseit és a hozzájuk fűzött megjegyzéseket

vagy átmozgatnunk az XOOPS modulkönyvtárba. Az én rendszeremen a WeBlog a `/usr/local/apache/html/xoops/modules/weblog` könyvtárba került.

A modul telepítéséhez, egyszerűen lépünk a Module Administration lapra és kattintsunk az install (telepítés) pontra. Különbféle weblog típusokat hozhatunk itt létre és állíthatunk be. A Module Administration lapon baloldalt megjelenik egy új napló-ikon, mellyel beállíthatjuk a jogosultságokat, a kategóriákat, valamint elvégezhetünk néhány alapvető adatbázis karbantartási feladatot.

Miután engedélyeztük a weblogok létrehozását, térjünk vissza a lapunk nyitóoldalára. Itt a képernyő bal oldalán már látnunk kell a WeBlog menüpontot a rendszer valamennyi legfrissebb weblog elemének gyűjteményével együtt. Ha a modult most futtatjuk első ízben, akkor ez a lap többnyire üres lesz. Saját weblogunkat a My WeBlog pontra kattintva hozhatjuk létre, illetve szerkeszthetjük. Küldeményünket gépeljük a HTML űrlap megfelelő részébe, legalább 75 karakter hosszúságban, hacsak a *Preferences Administration* lapon meg nem változtattuk ezt az értéket. Miután bevittük az új bejegyzésünket, a WeBlog menüelemre visszatérve meg is tekinthetjük azt. Ez a közös képernyő az összes egyéni naplót egyesíti, egyúttal azt is lehetővé teszi, hogy a felhasználók nevére kattintva, csak a kiválasztott blogot tekintsük meg.

A WeBlog számos egyéb hasznos lehetőséget is tartalmaz még, az RSS hírügynökségtől kezdve, a kereshető és böngészhető archívumon át a küldeményekhez fűzhető megjegyzésekig. Azt is kérhetjük, hogy az új küldemények felkerülésekor kapjunk figyelmeztetést. Az ilyen figyelmeztetések a főmenüben jelennek meg, amikor belépünk az oldalra.

### Használjuk az XOOPS-t?

Amennyiben blogot használó emberek hálózati közösséget szeretnénk létrehozni, az XOOPS nem rossz választás. Aktív fejlesztőcsapat támogatja, a WeBlog modul elég stabilnak és jól karbantartottnak tűnik. Továbbá a technológia egyszerű és magától értetődő ahhoz, hogy a web/adatbázis tapasztalattal rendelkezők maguk módosíthassák, állíthassák be saját rendszerüket.

Az XOOPS vizsgálgatása közben azonban nem bírtam megállni, hogy össze ne hasonlítsam azt az OpenACS-el, ezzel a szintén közösség centrikus rendszerrel, amely ugyanúgy rendelkezik weblog modullal. Az érdekelt, hogy vajon van-e bármilyen ok, amiért valaki inkább az XOOPS rendszerét választaná az OpenACS helyett, amely technikailag kétség-telenül messze fejlettebb.

Bármennyire is szeretem az OpenACS-et, a válasz egyértelműen igen. Az OpenACS kiváló rendszer ha egy keményvonalas web/adatbázis-guru van a közelben, és ha root jogosultságunk van a kiszolgálónkhoz, továbbá, ha az OpenACS eszközkészletet saját igényeinkhez tudjuk igazítani. Sokaknak azonban nincs megfelelő embere vagy pénze egy ilyen magasszintű rendszer használatához; ők a PHP-t és MySQL-t futtató megosztott kiszolgálókon is gyorsan és könnyen létrehozható megoldásokat keresik. Nekik az XOOPS nagyon jó választás. Ugyan sok minden hiányzik belőle, de előnyei is számosak: bárhova feltelepíthető, a PHP/MySQL programozók hatalmas közössége támogatja és minimális ismerkedés után a nem-programozók által is karbantartható. Sok dolog hiányzik belőle, amit egy kifinomultabb rendszertől elvárnék. Például nincs Unicode támogatása, nincs lehetőség egyetlen csomag több példányban való telepítésére, tartalom-változtatkezeltésre és hierarchikus csoportok létrehozására. De arra amire tervezték nagyon is jól használható.

Úgy érzem nehéz lenne a XOOPS-t a hagyományos értelemben vett tartalomkezelők közé sorolni, akármit is állítsanak a csomag karbantartói. Igaz, valóban lehetővé teszi, hogy új küldeményeket postázzunk és azokat a hely gazdája megjelenés előtt engedélyezze. Azonban az XOOPS rendszert inkább egy könnyen használható közösségi felületként képzelhetjük el, melynek segítségével egyszerűen hozhatunk létre weblogokat és egyéb szabványos tartalmakat. Sok embernek ez több is, mint elég. Az, hogy ilyen egyszerűen hozhatunk létre egy kifinomult honlapot, sokkal nagyobb súllyal esik latba mint az egyéb elvárások.

### Összefoglalás

Az XOOPS egyértelműen nem való mindenkinek – a világnézet, az objektum-központú kód és az említett funkciók hiánya miatt kétszer is meggondoltam, egyáltalán ajánljam-e. Csakhogy azt is láttam, mennyi időt és energiát emészt fel az indulás olyan alkalmazásokkal mint a Zope vagy az OpenACS – és ez a csere nem mindig kifizetődő. Ha kiválóan értünk a PHP és MySQL eszközökhöz, és jó weblog képességekkel felvértezett közösségi oldalt szeretnénk, nem árt elgondolkodni az XOOPS használatán.

A cikkhez tartozó források a CD-n a Magazin/XOOPS könyvtárban találhatóak.

*Linux Journal* 2004. június, 122. szám



**Reuven M. Lerner** (✉) <http://www.lerner.co.il/atf>

Nyílt forrású programokra, valamint web- és adatbázis-alkalmazásokra szakosodott tanácsadó. Könyve, a *Core Perl*, 2002 januárjában jelent meg a Prentice Hall gondozásában. Reuven feleségével és lányaival nemrég költözött Chicagóba.