

## A Subversion vállalkozás

Ismerkedjünk meg a hatékonyabb változatkezelő rendszer elképzelésével és megvalósításával.

**E**vállalkozás a CVS-rendszert (Concurrent Versions System) kiváltó programcsomag megalkotását tűzte ki célul. Aki valamilyen nyílt forráskódú program fejlesztésén munkálkodik, minden bizonnyal dolgozott már a CVS-sel, és talán még azt is fel tudja idézni, hogyan tanulta meg egy forráskódú névtelen ellenőrzését a Világhálón keresztül, az első programfordítást, vagy éppen az első CVS-változások figyelemmel kísérését. Aztán egyszer csak elérkezett a végzetes nap, amikor megkérdezte az egyik barátját, hogyan is kell átnevezni egy állományt: – Ó, azt nem lehet! – hangzott a válasz. – Micsoda? Hogy érted azt, hogy nem lehet? – kérdezett vissza. – Szóval, az állományt csak törölni tudod, majd új néven felvenni a nyilvántartásba. – Jó, akkor kérdezzük meg a CVS-gazdát! Vajon ő képes-e kézzel módosítani a CVS-rendszer RCS-állományait és képes-e újra működőképesé tenni a rendszert? – Jaj, csak most jut eszembe, hogy könyvtárt se próbálj meg törölni! Az ember a szemét forgatta és egyre csak sóhajtozott. Hogyan lehet az ilyen egyszerű dolgok végrehajtása ennyire bonyolult?

### A CVS öröksége

Kétség sem férhet hozzá, hogy a CVS a Nyílt Forráskód Közösségének programbeállító kezelőrendszerévé (SCM) fejlődött, és tegyük hozzá, joggal vált azzá. A CVS olyan nyílt forráskódú program, amelynek gördülékeny fejlesztési modellje az együttműködést egymástól nagy földrajzi távolságokra levő programozók százai számára teszi lehetővé. Sőt van, aki még azt is megkérdeje, hogy az olyan webhelyek, mint a Freshmeat vagy a Sourceforge a CVS nélkül képesek lettek volna-e úgy felvirágozni, mint amilyen sikerrel a mai napig működnek. A CVS és a hozzá tartozó károszerű fejlesztési modell mostanra a Nyílt Forráskód kultúrájának elemi részévé vált. Akkor mi a baj a CVS-sel? Mivel a CVS a háttérben az RCS tárolási

rendszert használja, csak az állományok tartalmi változásainak követésére képes, a faszervezetbeli változásokra nem. Ennek következtében a felhasználó számára nincs lehetőség az előzmények elvesztése nélkül az állományok másolására, áthelyezésére vagy átnevezésére. A faszervezetbeli átrendezések minden esetben csúnya kiszolgálóoldali csűrészavarásokat jelentenek. Az RCS a kiszolgálóoldalon képtelen a bináris állományok hatékony tárolására, az elágazási és jelölési műveletek igen lelassulnak. De a CVS a hálózatot is gyenge hatékonysággal használja, mint ahogy számos felhasználó bosszankodik a hosszú várakozási idők miatt: ezek abból adódnak, hogy az állománybeli módosulások küldése egyirányú – a kiszolgálótól az ügyfél felé és sohasem fordítva –, és a bináris állományok mindig teljes terjedelmükben kerülnek átvitelre. A fejlesztő szempontjából a CVS-alapok tulajdonképpen a történelmi javítások egymásra helyezett rétegei. Gondoljunk csak arra, hogy a CVS életének kezdeti szakaszában az RCS-t használó héjprogramok tömkelege létezett. Ez az örökség a programkódok értelmezését, karbantartását és kiterjesztését nehezkesé teszi. Például a CVS hálózati képessége csak „oda lett ragasztva”, de valójában sohasem volt igazi kiszolgáló-ügyfél rendszer. A CVS rendszer gondjainak megoldása óriási feladat, és ekkor még csak néhány gyakori panaszt említettünk.

### Lépünk be a Subversion rendszerbe!

1995-ben *Karl Fogel* és *Jim Blandy* megalapította a Cyclic Software nevű céget, amely a CVS-rendszer kereskedelmi alapú fejlesztésére és a felhasználók számára nyújtott terméktámogatásra vállalkozott. A Cyclic megjelenítette a CVS első hálózatképes, nyilvános változatát. 1999-ben Karl Fogel könyvet adott ki a CVS-ről és az általa lehetővé tett nyílt forráskódú programfejlesztésről (☞ <http://cvsbook.red-bean.com>). Karl és Jim már régen tervbe vették a CVS kiváltását; Jim még egy új raktárrendszer (repository) tervét is felvázolta.

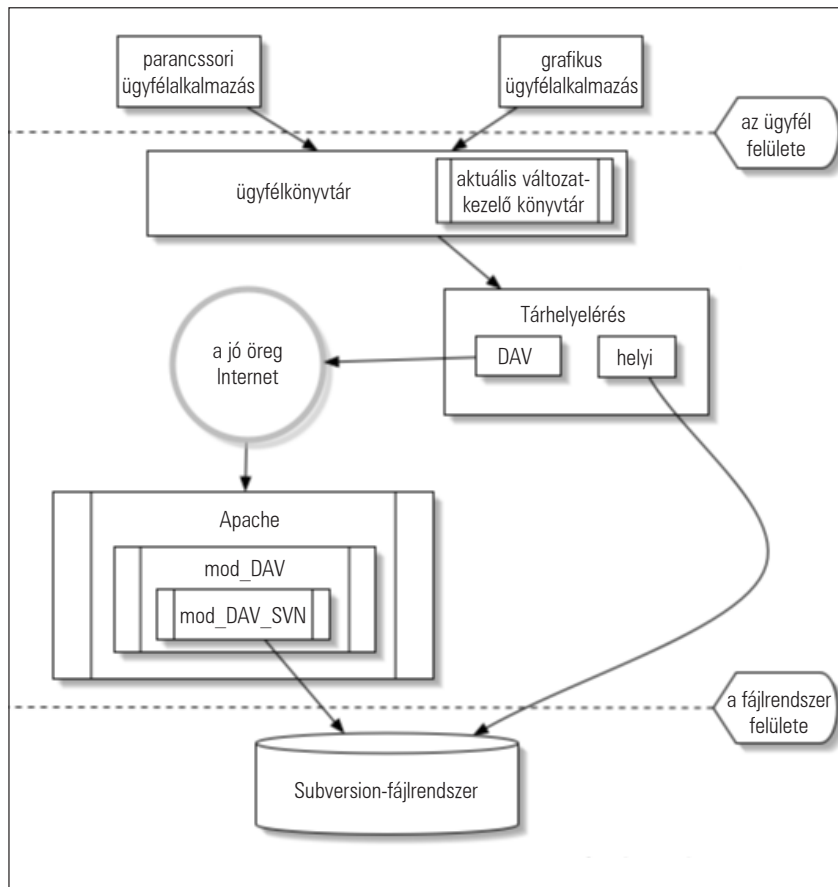
Végül 2000 februárjában a Collabnetnél (☞ <http://www.collab.net>) dolgozó *Brian Behlendorf* Karlnak teljes munkaidős állást ajánlott, és a CVS-rendszert kiváltó program elkészítésével bízta meg. Karl kiválasztotta a munkatársait, a csapat összeállt, és májusban megkezdődött a munka.

A fejlesztőcsapat néhány egyszerű célt tűzött ki maga elé: eldöntötték, hogy a Subversionnak működés szempontjából helyettesítenie kell a CVS-t. Mindenre képes lesz, amit a CVS el tudott végezni – megőrizve ugyanazt a fejlesztési modellt, ugyanakkor megoldja a CVS tervezési (hiányosságokból fakadó) hibáit. A jelenlegi CVS-felhasználók a programfejlesztés célközönsége; bármelyik CVS-felhasználónak csekély erőfeszítés révén képesnek kell lennie elkezdeni a Subversion használatát. Bármilyen további szolgáltatást csak másodlagos jellegűnek minősítettek – legalábbis az 1.0-s változat elkészülte előtt. E cikk írásakor a programfejlesztés már több mint egy éve folyt és számos kiváló programozó vett benne részt önkéntesen végzett munkájával. A Subversion a CVS-hez hasonlóan nyílt forráskódú vállalkozás.

### A Subversion szolgáltatás

Az alábbiakban felsorakoztatjuk azokat az érveket, amelyek azt támasztják alá, hogy a Subversion megjelenését nagy örömmel üdvözölhetjük:

- **Valódi másolatok és átnevezések:** a Subversion raktárrendszere egyáltalán nem használ RCS-állományokat. Ezek helyett olyan virtuális állományrendszert alkalmaz, amely nyomon követi a faszervezetek időbeli változását (lásd alább). Az állományok és könyvtárak egyaránt változatszámokkal lesznek ellátva. És végre az ügyféloldali mv és cp parancsok pontosan úgy viselkednek, ahogyan az elvárható tőlük.
- **Atomi műveletek:** egy művelet vagy teljes mértékben bekerül a raktárrendszerbe vagy egyáltalán nem kerül be.
- **Fejlett hálózati réteg:** a Subversion



A Subversion tervezési rétegei

hálózati kiszolgálóprogramja az Apache, az ügyfél- és kiszolgálógépek a WebDAV (2) protokoll segítségével értenek szót egymással (lásd alább a *Subversion tervezése* című szakaszt).

- **Gyorsabb hálózati elérés:** a módosítások tárolására és mindkét irányú továbbítására bináris eltérő algoritmust használnak, tekintet nélkül arra, hogy az állomány szöveges vagy bináris típusú-e.
- **Állományrendszer-jellemzők:** minden egyes állományhoz vagy könyvtárhoz egy elosztástábla tartozik. Tetszőlegesen kiválasztott kulcs/érték párokat adhatunk meg és tárolhatunk a felhasználókra, a jogokra, az ikonokra, az alkalmazástulajdonosra, a MIME-típusokra, a személyes megjegyzésekre stb. vonatkozóan. Ez a felhasználók számára nyújtott általános szolgáltatás. A tartalomhoz hasonlóan az állomány-jellemzők is kapnak változatszámot. Némelyik tulajdonság kijelölése önműködő, ilyen például az állomány MIME-típusa, ami azt jelenti, hogy a továbbiakban már nem

nagyon kell a -kb kapcsolóra gondolnunk.

- **Bővíthető és átalakítható:** a Subversion nem cipel „történelmi” terheket, a rendszer közös használatú C-könyvtárakra épül, jól meghatározott alkalmazási felületekkel együtt. Mindez a Subversiont rendkívül jól karbantartható, más alkalmazások és nyelvek számára kitűnően használható rendszerre teszi.
- **Könnyű áttérés:** a Subversion parancssoros ügyfélprogramjának működése nagyon hasonlít a CVS-hez, a fejlesztői modell azonos, úgyhogy a CVS-felhasználóknak az áttéréssel nem lehet túl sok gondjuk. A cvs2svn raktárrendszer-átalakító program fejlesztése folyamatban van.
- **Szabad program:** a Subversion az Apache, illetve BSD-szerű, nyílt forrású felhasználási engedéllyel jelenik meg.

### A Subversion felépítése

A Subversion felépítése modulrendszerű, megvalósítása C-könyvtárak halmozán alapul. Minden egyes rétegnek jól meghatározott célja és felülete van.

Általában véve a kódfolym a diagram tetejéről indul és lefelé halad, és minden réteg csatolófelületet biztosít a felette levő réteg számára (*ábránkon*). Járjuk be ezeket a rétegeket az alsótól kezdve!

### A Subversion állományrendszere

A Subversion állományrendszere nem rendszermagiszintű állományrendszer, amelyet az operációs rendszerbe kell telepíteni – mint például a Linux ext2 állományrendszert, ehelyett inkább a Subversion raktárrendszerére utal. A raktárrendszer, minthogy egy adatbázis tetejére épül, amely jelenleg a Berkeley DB, tulajdonképpen *.db* állományok halmaza. A könyvtár viszont hozzáfér ezekhez az állományokhoz, és egy C alkalmazási felületet emel be, amely egy állományrendszert utánoz, nevezetesen változatszámokat is rögzítő állományrendszert. Ez azt jelenti, hogy a raktárrendszerhez hozzáférő programot írni olyan, mint más állományrendszer számára alkalmazói programfelület készíteni: a szokásos módon nyithatunk meg állományokat és könyvtárakat olvasásra és írásra. A lényeges eltérés az, hogy ez az állományrendszer sohasem veszít el adatot, a régi állományok és könyvtárak – mint afféle történelmi alkotások – megőrződnek. A CVS a kiszolgálóoldalon az RCS-állomány(ok) révén állományonkénti alapon módosítási számokat tárol, ezzel szemben a Subversion teljes fákat lát el változatszámokkal.

A raktárrendszerben végzett apró módosítások teljesen új állományrendszerfát hoznak létre egyszerű, de mindenre kiterjedő módosításszámlálóval.

A módosított állományok és könyvtárak lemezre íródnak, miközben a régebbi állományokról visszatölthető mentés (back-up) készül, és a legfrissebb változathoz képest csak a változásokat tárolja. A változatlan bejegyzésekre az elosztott tárolási rendszerben szerepelnek hivatkozások, tehát a raktárrendszer ilyen módon nemcsak állományokat képes változatszámokkal ellátni, de egymásba ágyazott könyvtárakat is. Végül meg kell említenünk, hogy a Berkeley DB-hez hasonló adatbázis használata révén a Subversion más fontos rendszerszolgáltatást is képes nyújtani: az adatok sértetlenségét, atomi írásműveleteket, helyreállíthatóságot és a háttérállományok gyors előállítását. A részletesebb tájékozódás érdekében keresse fel a <http://www.sleepycat.com> címet.

## A hálózati réteg

A Subversion rendszer mindenhol magán viseli az Apache nyomát. A Subversion középpontjában működő ügyfélprogram az Apache Portable Runtime (APR) hordozható futásidejű könyvtárat használja. Ez azt jelenti, hogy a Subversion ügyfélprogramja bárhol lefordítható és futtatható, ahol a HTTPD futni képes. Jelenleg ide értendő az összes Unix-változat, a Win32, BeOS, OS/2, Mac OS X, sőt talán még a NetWare operációs rendszer is.

A Subversion nem kizárólag az APR-től függ; a Subversion-kiszolgálót maga az Apache httpd démon testesíti meg. Jó-jó, de miért pont az Apache-ot választották? Végül is azért esett rá a választás, hogy ne kelljen újra feltalálni a kereket, azaz mindent nulláról kezdeni. Az Apache nyilvános forrású, komoly használatra szánt rendszer, amely már kiállta az idők próbáját, ráadásul még mindig bővíthető. Nagy a hálózati terhelhetősége, sok felületen képes futni, és még a tűzfalakkal is együtt képes működni. Számos hitelesítési protokollt használni tud. Hálózati adatsatorna (pipeline) és gyorsítár (cache) feladatok is rábízhatók, és a Subversion az Apache kiszolgálóként való felhasználásával mindehhez a szolgáltatáshoz ingyen jut hozzá. Szóval, miért is kellett volna mindent az elejéről kezdeni? A Subversion saját hálózati protokolljaként a WebDAV-ot alkalmazza. A DAV (Distributed Authoring and Versioning) – elosztott szerzői és változat-nyilvántartó rendszer – önmagában is nagy téma (☞ <http://www.webdav.org>), dióhéjban azt lehet elmondani róla, hogy a HTTP kiterjesztése, amely a Világhálón állományok olvasását, illetve írását és változatkezelését teszi lehetővé. A Subversion reméli, hogy meglovagolhatja e protokollt lassan növekvő támogatását; a Win32, Mac OS X és a Gnome legfrissebb állománybőngészőinek mindegyike képes ennek a protokollnak a használatára. Egymással való együttműködésül (remélhetőleg) már egyre inkább afféle ráadásá válik.

Azon felhasználók számára, akik csak helyi a lemezen használják a Subversion raktárrendszerét, az ügyfélprogram ugyanúgy működik, ám ilyenkor nincs szükség hálózatra. A raktárrendszer-hozzáférési réteg (RA) olyan leegyszerűsített alkalmazói felület, amelyet a DAV és az RA helyi elérést támogató könyvtárai valósítanak meg. Ez a könyvtárossított módosításkövető rendszer előnye és a CVS felett aratott óriási győzelem. A CVS egyfajta programot használ a

helyi és egy másikat a hálózati raktárrendszer elérésére. Esetleg úgy gondolja, hogy új hálózati protokollt lenne kedve készíteni a Subversion-rendszerhez? Ha a kérdésre igennel felel, készítse el az RA alkalmazói programfelületet alkotó új könyvtárat.

## Ügyfélkönyvtárak

Ügyféloldalon a Subversion munkaműsolaat könyvtára a rendszer működéséhez szükséges adatot különleges /SVN könyvtárakban tárolja, amelyek a cél szempontjából a CVS-rendszerben található /CVS nyilvántartó könyvtárakhoz hasonlítanak.

Egy átlagos /SVN könyvtár belsejébe vetett pillantás viszont a szokásosnál jóval többet árul el magáról a rendszeréről. A bejegyzések állomány XML-t tartalmaz, amely a munkaműsolaat könyvtár pillanatnyi állapotát írja le, és alapvetően ez az, amely a CVS-bejegyzések, a gyökér- és raktárrendszer-állomány szolgáltatásait egyesíti magában. Ezenkívül sok minden más is megtalálható itt, ami nincs jelen a /CVS könyvtárban, többek között a változatszámokkal ellátott tulajdonságok tárolási helyei – ezeket fentebb a Subversion által nyújtott szolgáltatásoknál metaadatoknak neveztük – és minden egyes eredeti állományhoz gyorsítár. Ez utóbbi szolgáltatás teszi lehetővé, hogy a helyi módosításokról és változat-újrászámozásról hálózati elérés nélkül is kimutatást tudjunk készíteni. A hitelesítési adatokat ugyancsak /SVN könyvtárakban tárolják, semmint holmi .cvspass-szerű egyszerű kis állományban.

A Subversion ügyfélkönyvtárra van bízva a legnagyobb felelősség. Az a tiszte, hogy ötvözze a munkaműsolaat könyvtár feladatait a rendszerraktár-elérés könyvtár teendőivel, hogy ezekkel bármilyen alkalmazásnak a legmagasabb szintű felhasználói felületet biztosítsa, amely módosítási-szabályozási műveletet szeretne végezni.

Vegyük például a `svn_client_checkout()` eljárást, amely tulajdonságként egy URL-t használ. Ezt az URL-t átadja a rendszerraktár-elérési könyvtárnak és megnyit egy hitelesített munkamenetet (session) egy bizonyos rendszerraktárral. Majd a rendszerraktártól kér egy adott fát, amelyet elküld a munkaműsolaat könyvtárba, ekkor a teljes munkaműsolaat a lemezre íródik – vagyis a /SVN könyvtárak és minden egyébe. Az ügyfélkönyvtárt olyanra tervezték, hogy bármilyen alkalmazás használ-

hassa. A Subversion forráskódja szabványos parancssori ügyfélprogramot tartalmaz, de az ügyfélkönyvtár tetejébe tetszőleges számú grafikus felületű felhasználói programot ültetni sem nehéz feladat. Remélhetőleg ezek a felhasználói felületek egy nap majd sokkal jobban fognak szerepelni, mint a jelenlegi CVS grafikus felületek, hiszen nem többek a CVS parancssori ügyfélprogram kéré telepített törékeny csomagolásnál. Ezenkívül a megfelelő SWIG-hozzárendeléseknek (☞ <http://www.swig.org>) az alkalmazói programfelületet nagyszámú programnyelv számára hozzáférhetővé kell tenniük: Java, Perl, Python, Guile stb. A CVS legyőzésénél sokat számít az, hogy mindenütt jelen van.

## A Subversion jövője

A Subversion 1.0 megjelenését jelenleg 2002 elejére tervezik. Az 1.0-s közzétett változat megjelenését követően a Subversion olyan bővítmények befogadására szemelték ki, mint az i18n-támogatás, intelligens összeválogatás, jobb változatkészlet-kezelés, ügyféloldali bedolgozó programok, a kiszolgálógépnél pedig továbbfejlesztett szolgáltatások. A kívánságlistán szerepel még néhány ötlet, amelyekben a rendszerraktár elosztása, majd arról másodpéldány készítése kerül szóba.

Végül hadd idézzünk egy gondolatot a Subversion Gyakran Ismétlődő Kérdéseiből (GYK): „Mi (még) nem próbálunk meg új utat törni az SCM-rendszerekben, sőt még csak meg sem kíséreljük utánozni az SCM-rendszerek legjobb tulajdonságait. A CVS-rendszer kiváltására törekszünk. Ha három év múlva a Subversiont a Nyílt Forráskód Közösségében az SCM-rendszerek szabványaként fogják számon tartani, a vállalkozás már sikerrel járt. De a jövő egyelőre még kódos és titokzatos. Végsősoron a Subversionnek ezt a helyet a saját érdemei alapján kell kiérdemelnie. Javítófoltokat viszont továbbra is szívesen fogadunk.

*Linux Journal 2002. február, 94. szám*

*Ben Collins-Sussman*

már 11 éve dolgozik programozóként és rendszergazdaként különböző intézményeknél. Ben jelenleg a Collabnetnél, a Subversion fő támogatójánál tevékenykedik. Honlapja a ☞ <http://www.red-bean.com/sussman> címen olvasható.