

Nézzük, mi is az a CVS!

A CVS (Concurrent Versions System) rendkívül összetett változatkövető és kezelőrendszer. A fejlesztés közben elletkező forráskódok rendezésére, változásainak követésére szolgál, és arra, hogy mindezt bármely felhasználó, illetve az arra jogosultak számára elérhetővé tegye.

Először is nézzük a CVS előnyeit és hátrányait!

Előnyei:

- A kód jogosulatlan felülírásának kiküszöbölése.
Ez az előny elsősorban akkor tekinthető valódi előnynek, ha a fejlesztők száma nagy, hiszen egy kisebb fejlesztőcsapatban ez a gond még megoldható. Ezen túlmenően akkor sem nagy nyereség, ha a forráskód maga annyira moduláris, hogy több fejlesztő biztosan nem nyúl ugyanahhoz az állományhoz.
- Változatkövetés.
A változatkövetés nagyobb projektek fejlesztésekor szinte mindig jókora segítség, hiszen így nemcsak az emberi tényezőt kell figyelembe venni, de még a gép is segít nekünk abban, hogy az egyes részkódok a megfelelő számozást kapják, így nyomon követségük és a fejlesztés dokumentálása nagymértékben könnyebbé válik.
- Naplózás a készítés lépéseiről.
A CVS naplózza, ha valaki hozzányúl a kódhoz, de mi magunk is képesek vagyunk a naplóhoz megjegyzéseket fűzni, így újfent csak a fejlesztés nyomon követése és dokumentálása lesz egyszerűbb.
- Kódváltozás figyelése.
A CVS figyel, ha egy kódrészlet megváltozik, jelzi, amikor egy forráskódról adatot kérünk, sőt, ezt a változást a megfelelő formátumban akár le is kérhetjük a CVS-ről, így egy egyszerű patch nevű programmal meglévő forráskódjainkat kézzel is javíthatjuk.
- Frissesség.
A CVS használatából adódóan mindig naprakész, friss kódokat tartalmaz, bár ezek a legtöbb CVS-kiszolgáló esetében fejlesztői változatok, ám a legújabb forráskódot ettől függetlenül mindig CVS-ből tölthetjük le.
- Biztonságos azonosítás.
A CVS képes azonosítani és különböző jogokkal felruházni a felhasználókat. A CVS többféle azonosítási és használati lehetőséget biztosít, ezekről azonban a későbbiek folyamán ejtünk szót.

Hátrányai:

- Viszonylag nehézkes kezelhetőség.
Maga a CVS rendkívül összetett feladata és biztonsági eljárásai miatt nem tartozik a könnyen kezelhető programok közé. Néha még a grafikus alkalmazások használata is nehézkes és időrabló (a gcvs, xemacs beépített CVS kezelője stb.).
- Támadások elleni alacsony megbízhatóság.
Azoknál a fejlesztőcégeknél, ahol a forráskód titkos vagy nem adható ki, az Internetre kitett CVS nem

nyújt megfelelő biztonságot, hiszen nekik nagyon nem ajánlott a CVS ilyen használata. Mindezek mellett a CVS nem az a könnyen feltérhető típus.

- Lassúság.
A CVS, ha többen használják, lelassulhat. Igaz, ez a jelenlegi internetsebesség mellett nem annyira feltűnő, de egy belső hálózaton már meglátszik, hogy a CVS „gondolkodik”.

Hol érdemes használni a CVS-t? Érdemes-e használni?

Természetesen a válasz határozottan *igen*, hiszen a CVS nagyszerű eszköz a fejlesztők számára. A közös kódot nem szükséges valakinek minden órában összefésülnie, nem kell vigyáznunk arra, nehogy felülírjuk a másik munkáját, mert a CVS ügyel rá. A CVS-t leginkább nyílt forrású projektek fejlesztésekor érdemes használni, hiszen a fejlesztők nagy száma mellé itt rendkívül kevés koordinátor társul, akiknek munkáját egy CVS-rendszer nagymértékben megkönnyíti. Természetesen nagycégeknél is érdemes megfontolni a CVS használatát, mert ha a belső CVS-t elzárják a külvilágtól, a belső fejlesztőcsapat munkája jelentős mértékben hatékonyabbá és gyorsabbá válik. Mindenesetre nem árt némi felkészülés, mielőtt egy fejlesztőcég bevezeti a CVS-t, mert a kezelése tényleg nem túl egyszerű, használata pedig néha nehézkes, a munka azonban néhány jó héjprogrammal megkönnyíthető. Kezdjük meg hát az ismerkedést a CVS-sel!

Honnan szerezhetünk CVS-t?

A legegyszerűbb, ha letöltjük a CVS honlapjáról

➔ <http://www.cvshome.org>.

Ezenkívül a CVS csomag bármely Linux-változatban megtalálható. Ha beszereztük, telepítsük:

```
apt-get install cvs
rpm -i cvs
```

A telepítés folyamán néhány kérdésre kell válaszolnunk, elsőként a repository helyéről érdeklődik:

Where are your repositories?

Alapértelmezetten a `/var/lib/cvs` van megadva, amely nem rossz döntés, ám ha más elképzelésünk van arról, hogy a CVS hol tárolja azokat a forráskódokat, amelyeket a fejlesztők töltenek fel, adjunk meg neki más helyet. Arra azért ügyeljünk, hogy lehetőleg minél több helyet biztosítsunk a CVS számára, mivel egy projekt akár egy pillanat alatt is hihetetlen méreteket érhet el.

Másodikként a `history` fájl forgatására kérdez rá:

Do you want the history files in your
➔ repositories rotated weekly?

Ha azt szeretnénk, hogy history fájlunkat a rendszer hente forgassa, válaszoljunk igennel, egyébként nemmel. Itt választhatjuk az individual lehetőséget is. Ha igennel válaszolunk, akkor a telepítő megkérdezi milyen időközönként szeretnénk forgatni a helyét. A forgatáshoz a `cron` vagy `anacron` démonra van szükségünk. A következő kérdés a `pserver` engedélyezésére vonatkozik:



Should the CVS pserver be enabled?
Ez a CVS alapértelmezett kiszolgálótípusa. Erről a későbbiekben még szó esik, de ha hálózaton keresztül szeretnénk elérni a CVS-t, és nincs SSH-nk, válaszoljunk igennel. Ezután a CVS-kiszolgáló már fent is van. A pserver inetd-ből fut, így ha engedélyeztük a pservert, érdemes elindítani. A pserver a 2401-es kapun várakozik a kapcsolódásra.

A CVS általános változói:

Ezek környezeti változók, rendszerünkől függően az alábbi formátumok valamelyikét kell használnunk:

- export VALTOZO="ØrtØk" ;
- setenv VALTOZO "ØrtØk"
- SET VALTOZO="ØrtØk" (DOS alatt).

A CVSROOT a számunkra érvényes CVS-főkönyvtár beállítása. A példában belátható CVS kiszolgáló üzemmódban működik.

```
CVSROOT=" :pserver:cvsuser@cvshost.hu :
↳ /CVS/root/k nyvtÆr"
```

Amikor a CVS kiszolgálóként dolgozik, más protokollokat is használhatunk (ebben az esetben a CVS_SERVER változóban adhatjuk meg a használni kívánt protokollt):

```
CVSROOT=" :ext:cvsuser@cvshost.hu :
↳ /CVS/root/k nyvtÆr"
```

Ekkor a CVS nem működik kiszolgáló üzemmódban.

```
CVSROOT="/CVS/root/k nyvtÆr"/
```

Még sokféle értéket adhatunk a CVSROOT-nak, erre később térünk vissza.

```
CVSROOT=":met dus:felhasznÆl nØv@
↳ gØpnØv:repository_k nyvtÆra"
```

CVS_CLIENT_LOG

A naplófájl neve és teljes elérési útja. Lehetőség nyílik egy olyan fájlnev megadására, amelybe a CVS azt naplózza, hogy az ügyfél éppen milyen műveletet hajtott végre, és arra milyen választ kapott. A naplófájl alapján a CVS létrehoz egy \$CVS_CLIENT_LOG.in és egy \$CVS_CLIENT_LOG.out fájlt. Értelemszerűen a .in-ben a kimenő üzenetek az ügyfél által küldöttek, míg a .out-ban a kiszolgálótól kapott üzenetek találhatók meg. Természetesen ezt a naplózást a CVS csak akkor használja, ha kiszolgáló üzemmódban használjuk.

CVSIGNORE

Azon fájlok listája, amelyek a CVS számára nem láthatók. A fájlokat szóközzökkel kell felsorolni.

```
CVSIGNORE="also.txt
↳ masnaknem.c proba.c proba.h"
```

CVSREAD

Ha ennek a változónak értéke van, a CVS a letöltött fájlokat csak olvashatóra állítja, amelyeket így nem tudunk módosítani, amíg írható-olvashatóra át nem állítjuk őket.

CVSUMASK

Ezzel a változóval tudjuk beállítani a CVS-be került fájlok hozzáférési jogait.

EDITOR | CVSEEDITOR

Megadhatjuk kedvenc szövegszerkesztőnket.

A cvseditor felülbírálja az editor változó beállítását. A szerkesztőt a megjegyzések naplóba írásához használjuk (lásd később). Ha egyik változónak sem adtunk értéket, az alapértelmezett szövegszerkesztőt használjuk, ami legtöbb esetben a vi vagy a nano, esetleg az ae.

CVS_PASSFILE

Ebben a változóban írhatjuk felül a CVS jelszófájljának alapértelmezett helyét: \$HOME/.cvspass. Itt tárolja a CVS a különböző CVSROOT-okhoz adott jelszavakat.

CVS_SERVER

Ebben a változóban tudjuk a CVS-ügyfélnek megadni, hogy a kiszolgáló, amit el szeretnénk érni, milyen eljárást használ a hálózati kapcsolattartásra. Alapértelmezettként a cvs érték van megadva, amit érdemes ssh-ra módosítani – így az ügyfél is a jól ismert SSH-protokollt fogja használni. Természetesen ehhez az ügyféloldalon az SSH-ügyfélre (ssh), a kiszolgálóoldalon pedig az SSH-kiszolgálóra (sshd) van szükség.

A CVS változóit héjunk .rc fájljában érdemes beállítani, hiszen ha sokat dolgozunk CVS-sel, kényelmetlen lenne mindig újra és újra beállítani. Mindenképpen figyeljünk ezekre a változókra, mivel rengeteg felesleges munkát és időpocsékolást takaríthatunk meg.

Maga a CVS arra szolgál, hogy egy nagyobb fejlesztőcsapat munkáját összehangolja, figyeljen a különböző állapotok, szálak változataira, forráskódjára. A CVS segíthet nekünk a kód rendszerezésében, a leírás elkészítésében és a zökkenőmentes munka összehangolásában. Néhány program, amit CVS-ben fejlesztenek:

- PHP, a közkedvelt parancsnyelv,
- MPlayer, a linuxos filmlejátszó,
- Wine,
- gcc, a GNU C fordító,
- CVS, a változatkövető rendszer.

A CVS előnyeivel és hátrányaival együtt a legjobb és legmegbízhatóbb változatkövető rendszer, így használata melegen ajánlott – legyen rendszerünk akár Linux, akár BSD, netán DOS.

Vass Endre

PHP-programozóként dolgozik, szabadidejében éppen egy számítógépes szerepjátékot fejleszt C/CC++ nyelven. Szereti a régi számítógépeket, kedvenc változata a Debian GNU/Linux.

