

Fájlszinkronizálás Unisonnal

Több gépen egyszerre szinkronban tartani a könyvtárakat nem könnyű feladat. Az egyik lehetőség amivel egyszerűbbé tehetjük életünket, az Unison használata.

■ Az *Unison Linux*, *UNIX* és *Microsoft Windows* környezetben futó fájl szinkronizáló program. Akik már használtak *IBM Lotus Notes* vagy *Intellisync Mobile Suite* programokat nagyjából tudják miért is jobb a szinkronizálás mint az olyan egyirányú tükrözési megoldások mint az *rsync*.

Elképzelhető például, hogy a cég dokumentum könyvtárát a laptopunkra tükrözzük, majd módosítunk egy-két dokumentumot. Mások ugyanebben a könyvtárban más állományokat módosíthattak, mire visszaérünk. Az *rsync* esetében kézzel kellene elsimítanunk a két könyvtár közötti különbségeket vagy kockáztatjuk, hogy felülírjuk valakinek a munkáját. Az *Unison* ki tudja válogatni mi és hol változott, terjeszti a megváltozott állományokat sőt, ha megmondjuk miképpen tegye, akár az azonos fájlok változásait is képes összeolvasztani.

Az *Unison*ot úgy is elképzelhetjük, mint egy kis változáskezeléssel fűszerezett kétutas *rsync* programot. Leggyakoribb felhasználási területe a helyi és távoli saját könyvtár vagy egy különféle környezetekben gyakran használt könyvtár szinkronizálása. Az *rsync* algoritmusával tartja alacsonyban a hálózati forgalmat és *SSH*-n keresztül vihető át a megbízhatatlan hálózatokon. Mindez többletmunkát nem jelent számunkra: egyszerűen a *ssh://* kifejezést kell megadnunk amikor a helyi könyvtárat bejegyezzük. Ugyanakkor az *Unison* használata során elég szép mennyiségű extra helyre lesz szükségünk, ugyanis a szinkronizáló algoritmusnak tudnia kell, hogyan néztek ki az állományok a legutóbbi futtatáskor.

Unison beszerzése, fordítása és telepítése

Az *Unison* honlapja az *Pennsylvania Egyetemen* található; a projekt vezetője, *Benjamin C. Pierce*, a *Számítástechnika és Információtudomány Tanszékének* professzora. Az *URL*-t a források közt találjuk.

Az *Unison*ot nem alkalmazzák olyan széles körben mint az *rsync*-et, így előfordulhat, hogy a terjesztésünkhöz nem találunk előre fordított csomagot. Az *Unison* honlapról letölthető binárisok azonban valószínűleg a legtöbb embernek megfelelőek lesznek.

Ha forrásból szeretnénk lefordítani, ezt is megtehetjük. Igaz ez esetben pár körrel többet kell futnunk, ugyanis az *Unison* a nem túl gyakori *Ocaml* nyelven készült.

Ha terjesztésünk nem tartalmaz kézhez álló csomagot, a források közt megtaláljuk.

1. lista .unison/default.prf

```
# Az Unison beállító állomány
merge = diff3 -m CURRENT1 OLD CURRENT2 > NEW
backup = Name *
maxbackups = 10
log = true
logfile = /home/knan/.unison/unison.log
rshargs = -C
```

Az *Unison* fordítása és telepítése egyszerű; adjuk ki a `make UISTYLE=xxx` parancsot. A *GTK* felhasználói felülethez további *OCaml GTK* bővítményekre lesz szükségünk, ezért cikkünkben a szöveges felületre szorítkozunk. A `make UISTYLE=text` vagy `make UISTYLE=gtk` parancs segítségével kapjuk meg az *Unison* végrehajtható állományt. Egyszerűen másoljuk az állományt valamilyen elérhető helyre mindkét szinkronizálandó gépen.

Cikkünkben ha külön nem említjük, az *Unison* legfrissebb 2.9.1-es stabil változatát használjuk. Amennyiben 2GB-nál nagyobb állományokat szeretnénk szinkronizálni, jelenleg a béta változatot kell alkalmaznunk.

Általában a fejlesztői verziók is jól működnek. A fejlesztők is ezeket használják saját értékes adataikon. A kalandvágyóak feliratkozhatnak az *Unison* programozók levelezőlistájára is. *Jerome Vouillon*, *Benjamin C. Pierce* és *Trevor Jim* általában itt vitatják meg a fejlesztéseket. Gyakran láthatunk küldemény-naplókat is, úgyhogy nyomon követhetjük a történéseket.

Unison beállítása és használata

Az *Unison* saját beállításait és munkafájljait a *.unison* könyvtárban tartja a saját könyvtárunkon belül vagy ahová éppen tenni szeretnénk. Az *UNISON* környezeti változót beállítva más helyet is megadhatunk.

Az alapértelmezett beállításokat a *.unison/default.prf* állomány tárolja. Az *1. lista* teszteléshez használható egyszerű beállítást mutat. Két könyvtár szinkronizálása ezek után mindössze ennyiből áll:

```
$ unison /nfsmount/dir1 /home/me/dir1
```

2. lista .unison/home.prf

```
# Unison beállítás állomány
root = /home/erik
root = ssh://remotehost/home/erik
# vagy pontosan kettő "root" sor lehet,
# vagy egy sem
ignore = Name *.mp3
# az összes .mp3 állományt figyelmen kívül
# hagyjuk
ignore = Path .unison
# nem foglalkozunk az elérési útjukban .unison
# szót tartalmazó # állományokkal
include default
# az alapbeállításokat a default.prf-ből
# olvassuk
```

Az **Unison** ezek után megkérdezi, van-e különbség a könyvtárak között majd elfogadható alapértékeket kínál fel. Beletelhet kis időbe míg hozzászokunk az **Unison** gondolkodásmódjához. Fontos, hogy az **Unison** nem helyettesíti a mentéseket. Például nyugodt szívvel tovaterjeszti az egyik másolatban bekövetkezett fájlterlekeket, ami hidegzuhanyként érheti a CVS-hez szokott programozókat. Például:

```
rm dir1/* ; unison ssh://server/dir1 dir1
```

egyáltalán nem úgy viselkedik mint a következő sortól várjuk:

```
rm dir1/*; cvs update dir1
```

A törlés ugyanolyan művelet, amit le kell másolni szinkronizáláskor. A példaparancs ennek megfelelően mindkét oldalon törli a dir1 könyvtár tartalmát. Miután kicsit belerázódtunk, érdemes az auto = true utasítást elhelyezni az **Unison** állományába. Így az ütközést nem okozó változásokra nem kérdez rá a rendszer de a végén lehetőséget ad a visszalépésre. Érdemes áttanulmányozni az **Unison** kézikönyvét. Tisztán és világosan fogalmazott és jól rávilágít mi fog történni az egyes határesetekben .

Sajátkönyvtárak szinkronban tartása

Amikor a felhasználó megbarátkozott az **Unisonnal** gyakran felmerül a sajátkönyvtárak gépek közötti szinkronizálásának kérdése, például a laptop és az asztali gép között. Ezt nagyon könnyen megvalósíthatjuk. A 2. listában olvasható egyszerű beállítások ezt a feladatot oldják meg, bár valószínűleg bővíteni akarjuk majd. A 2. lista, például, figyelmen kívül hagyja az MP3 állományokat és az **Unison** saját állományait valamint bemutatja az include működését mellyel valamennyi profilon azonos beállításokat érvényesíthetünk.

Új profilunkat a következőképpen próbálhatjuk ki:

```
$ unison home -testserver
```

Meghívása pedig így történik:

```
$ unison home -batch
$ unison home
```

A -batch futtatás kezeli az összes egyszerű esetet visszakerdezés nélkül, ment és naplóz szükség szerint, majd a második futtatásnál az összes trükkös dologra rákérdez – például az összeolvasztásokra.

A root = sorok elhagyhatók ha inkább a parancssorban szeretnénk megadni a szinkronizálendő állományokat.

A sorok a következő parancshívásnak felelnek meg:

```
$ unison home /home/erik
↳ ssh://remotehost/home/erik
```

Ütköző változások összeolvasztása

A három utas összeolvasztás kivitelezéséhez be kell kapcsolnunk a mentéseket. Alapértelmezés szerint a háttérmentés nincs engedélyezve, az **Unison** csak az ellenőrző összegeket és metaadatokat (például a jogosultságokat) tárolja tehát nincsen felhasználható módosíthatatlan változata. Az **Unison 2.9.1**-es változatában, ha az ütközés feloldása érdekében kiadott összeolvasztás művelet kézi beavatkozás nélkül is sikeres volt, a változások azonnal szétterjednek, ami nem ad lehetőséget a visszavonásra. Ezért, ha van elég helyünk, javasolom a maxbackups értéket hagyjuk legalább 5-ön az alapértelmezett 2 helyett, így megadjuk az esélyt, hogy az esetleges hibás összeolvasztásokat helyre tudjuk állítani. A mentési könyvtár összeolvasztás után a következőképpen néz ki:

```
$ ls -l .unison/backup/
shared.txt           összeolvasztott verzió ("NEW")
shared.txt.1.unibck távoli változat ("CURRENT2")
shared.txt.2.unibck helyi változat ("CURRENT1")
shared.txt.3.unibck régi verzió ("OLD")
```

A legfrissebb **Unison** béta változat (írásunk születésekor a 2.10.3) eltérő állományokhoz eltérő összeolvasztó programokat képes meghívni. **XML** állományok összeolvasztásához például valószínűleg szívesebben használnánk **3DM**-et míg a **Berkeley** adatbázisokhoz a valamilyen adatbázis olvasztó eszközt. Ez a képesség még új és képlekeny. A projektvezető meg is jegyezte, hogy az összeolvasztó képességet mindenképpen újra kellett írni és nem működött igazán jól a 2.9.1 és 2.9.20 változatban. Ezért ha sok olvasztást szeretnénk végezni, jobban járunk ha a legfrissebb változatokból szemezgetünk.

Linux Journal 2005. április, 132. szám



Erik Inge Bolso UNIX tanácsadó és épée vívó a Norvégiai Molde-ban, és 1996 óta futtat Linuxot. Másik hobbijáról a „balrog genealogyról” egy Google keresés segítségével kaphatunkinformációt, őt magát pedig a ljcomment@tvilsom.org címen érhetjük el.

KAPCSOLÓDÓ CÍMEK

A cikk forrásai: ➔ www.linuxjournal.com/article/8059