



Bisoños Usuarios de GNU/Linux de Mallorca y Alrededores | Bergantells Usuaris de GNU/Linux de Mallorca i Afegitons

Darcs: control de versions redux (17732 lectures)

Per Pau Rul-ian Ferragut, [paurullan](http://bloc.baleaerweb.net/paurullan) (<http://bloc.baleaerweb.net/paurullan>)

Creado el 21/08/2006 00:52 modificado el 21/08/2006 00:52



Darcs⁽¹⁾ és un [sistema de control de versions distribuït](#)⁽²⁾. Aquest concepte juntament d'un desenvolupament teòric anomenat teoria de pedaços fan d'ell un sistema molt a tenir en compte.

Presentació

Característiques de darcs

Si volguessim enumerar les carecterístiques de darcs podríem dir:

- Possibilitat de fer commits locals (sense connexió)
- Se poden desfer commits concrets (el que es coneix com *cherry picking*)
- Cada repositori és una branca per si mateixa
- Independència d'un servidor central
- Simetria als repositoris (tots segueixen els mateixos esquemes)
- Possibilitat de renombrar fitxers mantenint l'historial
- Podem definir paràmetres a nivell d'usuaris o de repositoris
- Simpleza infinita a l'hora de posar en marxa un repositori
- Suficientment documentat (i sinò [sempre teniu el codi font](#)⁽³⁾)
- Varis mètodes d'accés: local, ssh, http i ftp
- Suport per fitxers binaris

El major inconvenient de la implementació actual de darcs és l'ús de memòria. Aquest és el punt on s'estan posant els esforços de desenvolupament però la situació actual no és satisfactoria per necessitats a gran escala (estic parlant de repositoris de més de dos-cents cinquanta megas en fitxers). Quan feim el checkout inicial tot el repositori puja ha d'estar en memòria i això és simplement un gran problema. La memòria física tampoc facilita molt: hem de tenir una còpia de tot el repositori i de tot l'historial. És a dir, quan feim una còpia del repositori no podem baixar-nos el head sinò que ens hem de menjar-nos-ho tot. La conseqüència directe d'això és que projectes grossos com serien el kernel de Linux, els grans escriptoris o les X.org no es proposaran migrar a aquesta eina. Cal dir que existeix git, el programa que actualment usen els desenvolupadors del kernel, que malgrat no és exactament igual que darcs les directrius generals si que són molt semblants.

Nou concepte: control de versions descentralitzat

Després de l'hegemonia del CVS es creia que havia poc a fer amb el tema del control de versions. El conegudíssim Subversion, reemplaç lògic del CVS, simplement reimplementava el concepte dels vcs clàssics. Però no sempre un control de versions centralitzat és la resposta als nostres problemes. Aquí és on entren els controls de versions *descentrelitzats*. Aquests sistemes no sols representen una important innovació al món de la enginyeria del programari sinò demanen un canvi de *xip* i una manera de fer les coses un poc distinta a lo que esteim avesats.

La peça clau de la descentralització del control de versions és que no existeix un únic repositori sinò que **tots** els repositoris estan al mateix nivell. Amb això vull dir que cada repositori, cada pull, és una branca distinta. D'aquesta



manera ens asseguram una gran independència entre desenvolupadors, donant moltes facilitats no sols al desenvolupador solitari i esporàdic sinò també al petit escamot de code-monkeys. Cal recordar que el fet que sigui descentralitzat no ens limita a seguir els mètodes i les tècniques dels sistemes centralitzats com Subversion. A més, amb el que es coneix com *teoria de pedaços* podem assegurar que en qualsevol moment podem ficar la nostra feina de la branca a un repositori que considerem *central*.

Posarem alguns exemples on la descentralització ens pot ser útil.

- Volem fer un commit i no tenim accés (per falta de connexió, per exemple). A mi m'agrada fer un nou record (sinònim de commit a la jerga de darcs) cada vegada que acabo d'implementar una nova funcionalitat que duu feina o quan he corregit un error greu.
- Acabam de publicar una versió estable del nostre programa i volem deixar tranquil el repositori central. Com que cada còpia és un repositori per si mateix no ens hem de preocupar per res.
- Esteim fora connexió i un nou vingut vol col·laborar. Ell pot fer un pull (sinònim de checkout a darcs) de la *meva* branca i no de la del repositori central. Aquesta situació se podria veure com una extensió del casos anteriors.
- Tenim un projecte de programari lliure on nosaltres som els principals desenvolupadors però volem acceptar col·laboracions externes. Podem posar un repositori de darcs d'on els usuaris potencials puguin agafar el codi i enviar-nos amb molta facilitat els canvis. Ens estalviem molta feina de gestió de permisos
- Un sistema distribuït pot significar un accelerador per la trobada i arreglo de falles. Evidentment es pot complementar amb alguna eina rollo bugzilla o mantis. La gràcia és que pel fet que tinguem l'estructura completa a la punta dels dits i que amb dues ordres poguem enviar a l'administrador del repositori els canvis ens estalviarem moltes passes a l'hora de corregir codi. I ja se sap que un desenvolupador content és un desenvolupador productiu.

En tot recordar que la descentralització del repositori és complementari al model centralitzar. Tot depèn de les nostres pràctiques i com l'usem però poder seguir amb la rutina clàssica si no ens sabem acostumar al darcs.

La peça clau: la teoria de pedaços

De l'anglès *patch theory*, la teoria de pedaços és tot un conjunt de proves i teoremes que permet la posada en pràctica dels sistemes de control de versions descentralitzats. Els tres conceptes importants són el de pedaça, arbre i contexte. L'**arbre** són el conjunt de fitxers i un **pedaç** és un canvi a l'arbre. Això du a que tot és un pedaça: afegir un fitxer, esborrar-lo, reanomenar-lo, canviar-lo un poquet o buidar el seu contingut. Quedau-vos amb això: *tot és un pedaça*.

El **contexte** és el que ajuda la definició de la **representació** d'un pedaça. Si la definició és el que fa únic i distint cada pedaça el contexte és el que determina on s'ha d'aplicar. Com que pot resultar molt complexe decidir on comença un arbre el que es fa és que l'estat inicial del repositori sigui un arbre buit. Per això fins i tot afegir fitxers és un pedaça.

El tema és molt interessant però tampoc vull entrar en detalls. Si voleu podeu seguir llegint sobre ell al mateix manual. Malgrat tot m'agradaria comentar dues propietats que facilitaran molt la vida a l'administrador.

La inversa d'un pedaça és el pedaça més senzill tal que la composició de l'original i el seu invers no fa canvis a un arbre.

Com a teorema afegim que

la inversa de la composició és la composició dels inversos.

Per fer terriblement senzilla la resolució de conflictes tenim un altre teorema que, grosso modo, explica que els *pedaços són commutatius*. D'aquesta manera ens evitam preocupar-nos de l'ordre en que apliquem els pedaços i, fins i tot algunes vegades, d'aplicar tot l'historial. Això significa que d'uns canvis a, b i c podríem sols aplicar a i c i no tindriem cap conflicte.

Exemple d'ús

La manera més clara d'aprendre a usar un sistema de control de versions és provar-lo. Per aquest motiu he preparat i deixat un repositori de darcs on tots vosaltres i podeu jugar. Jo també he anat fent-li coses i deixo aquí algunes de les



mostres.

Resum ordres

Sense ser un substitut del manual aquest troç podria ser una ràpida referència. Aquí en teniu [una per gent que ve de svn](#)⁽⁴⁾.

- help: l'utilíssima ajuda en línia.
- add: imprescindible si volem afegir material.
- remove: desvincula de l'arbre sense esborrar el fitxers. Si esborrèssim el fitxer directament també desapareixeria de l'arbre i quedaria constància del borrat.
- mv: mou, de veres i no com fa svn o cvs, fitxers.
- (un)revert: desfà o refà els canvis a l'arbre. Recordau que és el revert el que heu d'executar si heu esborrat accidentalment fitxers.
- (un)record: desa els canvis a un pedaç o aplica l'invers d'algun pedaç (és a dir, desfà els canvis del pedaç ja aplicat).
- whatsnew: mostra les diferències entre la situació actual i la darrera actualització.
- resolv: marca la resolució de conflictes.
- tag: deixa una marca a l'historial de l'arbre. En realitat és un record sense modificacions als elements.
- changes: mostra l'historial de canvis.
- dist: fa un tar.gz del trunk de l'arbre.
- send: envia els pedaços generats pel record al mantenidor del repositori.
- apply: aplicam algun pedaç que ens hagin enviat.
- pull: actualitza el repositori amb base al passat com a paràmetre. Aquesta ordre deixa marcat el darrer repositori, així que no farà falta tornar a escriure-lo si no canvia la localització.
- push: puja els canvis gravats al servidor pare.
- get: feim una branca del repositori objectiu.
- put: complementari del get. Ens ficam a un repositori i el paràmetre és la localització del nou repositori.

La sintàxi amb els protocols és senzilla però s'ha de saber:

- ssh: darcs (pull|push) nomdusuari@maquina:cami/al/repo
- http: darcs (pull) http://maquina/cami/al/repo
- ftp: darcs (pull) ftp://nomdusuari:claudepass@maquina:cami/al/repo
- ftp: darcs (pull) ftp://nomdusuari@maquina:cami/al/repo (demanarà contrassenya)
- correu: darcs (send|apply)

Aquest mateix article a un repo darcs

```
darcs initialize
echo "algun text per posar al nou fitxer" > primer-fitxer
darcs add -r .
darcs record -la
```

Amb això ja tenim el nostre repositori inicialitzat i amb al primer commit. Després de fer el record se'ns farà alguna pregunta com seria el correu electrònic de l'autor. Ara podem fer el pull d'una adreça http.

```
% mkdir apache-article-darcs
% cd apache-article-darcs
apache-article-darcs % darcs initialize
apache-article-darcs % darcs pull 'http://bulma.net/~paurullan/article-darcs/'
```

```
Sun Aug 20 20:25:43 CEST 2006 Pau Rul-lan Ferragut
* primer commit
Shall I pull this patch? (1/1) [ynWvpxqadjk], or ? for help: y
Finished pulling and applying.
```

```
ssh-article-darcs % darcs push
```



```
Pushing to "bulma.net:/home/socis/paurullan/public_html/article-darcs"...
```

```
Sun Aug 20 20:46:30 CEST 2006 paurullan@bulma.net
* afegit directori d'imatges
Shall I push this patch? (1/1) [ynWvpxqadjk], or ? for help: a
Finished applying...
```

Amb un accés per ssh tindrem els permisos que tenim a la màquina remota, així que si ho teni configurat podrem escriure.

```
% mkdir ssh-article-darcs
% cd ssh-article-darcs
ssh-article-darcs % darcs initialize
ssh-article-darcs % darcs pull\
bulma.net:/home/socis/paurullan/public_html/article-darcs/
Sun Aug 20 20:25:43 CEST 2006 Pau Rul-lan Ferragut
* primer commit
Shall I pull this patch? (1/7) [ynWvpxqadjk], or ? for help: a
Finished pulling and applying.
```

```
ssh-article-darcs % touch quart-fitxer
ssh-article-darcs % darcs add quart-fitxer
ssh-article-darcs % darcs record -al quart-fitxer
Recording changes in "quart-fitxer":
```

Darcs needs to know what name (conventionally an email address) to use as the patch author, e.g. 'Fred Bloggs '. If you provide one now it will be stored in the file '_darcs/prefs/author' and used as a default in the future. To change your preferred author address, simply delete or edit this file.

```
What is your email address? paurullan@bulma.net
What is the patch name? afegit quart fitxer
Do you want to add a long comment? [yn]n
Finished recording patch 'afegit quart fitxer'
```

```
ssh-article-darcs % darcs push -a
Pushing to "bulma.net:/home/socis/paurullan/public_html/article-darcs/"...
Finished applying...
```

Branques locals, part de la màgia dels repos descentralitzats. El comportament és igual a l'anterior sols que estarem vinculats al ssh-article-darcs envers al de la web.

```
% darcs get ssh-article-darcs copia-ssh-darcs
darcs get ssh-article-darcs copia-ssh-darcs
cd copia-ssh-darcs
touch segon-fitxer
darcs add -r .
darcs record
darcs push
```

Canviant els repositoris. Teniu en compte que això s'hauria de fer tocar els fitxers de configuració i fent el push amb el nou repositori com a paràmetre.

```
copia-ssh-darcs % touch tercer-fitxer
copia-ssh-darcs % darcs add tercer-fitxer
copia-ssh-darcs % darcs record -al
What is the patch name? afegit tercer fitxer
Do you want to add a long comment? [yn]n
Finished recording patch 'afegit tercer fitxer'

copia-ssh-darcs % darcs push
Pushing to "/home/paurullan/experiment/ssh-article-darcs"...

Sun Aug 20 21:21:55 CEST 2006 paurullan@bulma.net
* afegit tercer fitxer
Shall I push this patch? (1/1) [ynWvpxqadjk], or ? for help: a
```



```

Finished applying...

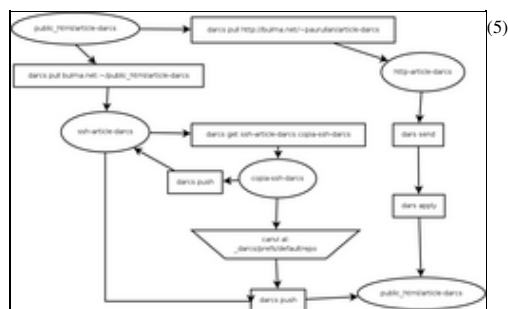
(ara podriem fer: copia-ssh-darcs % darcs push \
bulma.net:/home/socis/paurullan/public_html/article-darcs)
copia-ssh-darcs % echo "bulma.net:/home/socis/paurullan/public_html/article-darcs" >\
_darcs/prefs/defaultrepo
copia-ssh-darcs % darcs push
Pushing to "bulma.net:/home/socis/paurullan/public_html/article-darcs"...

Sun Aug 20 21:02:30 CEST 2006  paurullan@bulma.net
* segon fitxer
Shall I push this patch? (1/2)  [ynWvpxqadjk], or ? for help: a
Finished applying...

```

Aquest seria un esquema del que hem fet amb els repositoris. Fixau-vos com els cercles són repositoris per si mateixos. Les capses són una pista de les ordes que hem executat durant l'exemple. Recordau que el canvi d'objectiu al repositori l'hem fet a mà i que seria el mateix que fer un `push http://bulma.net/~paurullan/article-darcs/`. També és important notar que si volem tornar a actualitzar el nostre `copia-ssh-darcs` haurem de fer un `pull` i no un `get`.

Podriem seguir però aquest article no té la intenció de ser un manual per conèixer els repositoris. Igualmente i per suposat si voleu fer millores a aquest article ja sabeu que teniu un arbre darcs sols per ell ^_^



Podreu fer clic sobre la imatge per veure-ne un de més gran.

Referències

Un parell de planes que vos poden ser útils.

- [Taula comparativa de sistemes de control de versions](#)⁽²⁾
- [Pàgina oficial del programa](#)⁽¹⁾
- [Wiki del programa](#)⁽⁶⁾
- [Interessant article a la LinuxWeeklyNews](#)⁽⁷⁾

Lista de enlaces de este artículo:

1. <http://abridgegame.org/darcs/>
 2. http://zooko.com/revision_control_quick_ref.html
 3. <http://recurrente.afraid.org/myblog/?q=node/134>
 4. <http://darcs.net/DarcsWiki/MigratingFromSubversion>
 5. <http://bulma.net/~paurullan/article-darcs/imatges/diagrama-repositoris.png>
 6. <http://darcs.net/DarcsWiki>
 7. <http://lwn.net/Articles/109719/>
-

E-mail del autor: paurullan_ARROBA_bulma.net

Podrás encontrar este artículo e información adicional en: <http://bulma.net/body.phtml?nIdNoticia=2333>