

Packaging

Compilation au préalable

- Connexion au compte **a-sac-ns-brainvisa**

Les compils sur les branches trunk et bug_fix sont faites toutes les nuits, mais il peut être utile de relancer la compil d'abord :

- [REDACTED]
- [REDACTED]

Cette commande commence par faire une mise à jour du dépôt subversion.

Machines pour les compils

voir maintenant: https://bioproj.extra.cea.fr/redmine/projects/catidev/wiki/Nightly_builds_in_Neurospin_using_Casa-distro

Linux		
System	OSID	Name / IP
Fedora 4 32 bits	Fedora-4-i686	obsolète - ne compile plus. 132.166.142.55 (VM i2bm-fdr4-32)
Mandriva 2008 64 bits	Mandriva-2008.0-x86_64	132.166.142.101 (VM i2bm-mdv-64)
Ubuntu 16.4 64 bits	Ubuntu-14.6-x86_64	is234199 (machine de Denis, older: is208611)
Ubuntu 14.4 64 bits	Ubuntu-14.4-x86_64	VM i2bm-ub1404
Ubuntu 12.4 64 bits	Ubuntu-12.4-x86_64	VM i2bm-ub1204
CentOS 5.3 64 bits	CentOS-5.11-x86_64	michael (GIPSI machine)
MacOS		
MacOS 10.5 32 bits	MacOS-10.5-i386	is144451 (bureau de Denis)
Windows		
Windows 7 32 bits	Windows-7-i686	cross-compil depuis Docker / Ubuntu 14.04, lancé depuis is208614 (machine de Yann)
Windows 7 64 bits	Windows-7-x86_64	cross-compil depuis Docker / Ubuntu 14.04, lancé depuis is208614 (machine de Yann)
Windows XP 32 bits	Windows-XP-i686	obsolète - ne compile plus. (VMs i2bm-xp-32 / i2bm-xp-64)

Pour se connecter à la VM Windows, on peut utiliser **tsclient** ou **remmina** ou **grdesktop**. Mais attention, ça déconnecte les autres personnes éventuellement connectées dessus.

Le script `bv_build_nightly`, qui lance la compilation de tous les projets, est lancé toutes les nuits sur ces machines grace à une crontab. Pour modifier la crontab, il faut se connecter sur la machine avec le compte `a-sac-ns-brainvisa` et faire `crontab -e`.

Création des pack sur les différents systèmes

Note: Avant de faire un package, il faut avoir bien pris soin de compiler complètement les projets, et aussi d'avoir **génééré leur documentation** (`make doc`, ou `bv_maker doc`), sans quoi il y aura des erreurs incompréhensibles pendant la création des packs.

Le script `bv_packaging_i2bm` permet de faire les packages complets BrainVISA, il utilise le script `bv_packaging` auquel il passe en

paramètre la liste des projets à packager. Il génère également le nom du package en `brainvisa-<system>-<version>-<date>` et le compresse.

Attention : `bv_packaging` crée le package à partir du répertoire de build qui se trouve dans les path. Ne pas oublier d'utiliser `bv_env.sh` pour se positionner sur le bon répertoire de build avant de lancer le packaging.

Pour Linux, `bv_packaging` appelle également un script `bv_patch_libs.sh` qui permet de supprimer les RPATH potentiels dans les headers des bibliothèques packagées pour éviter que les bibliothèques du pack n'aille chercher des bibliothèques du système. Pour que ce script fonctionne, il faut que l'un des outils `chrpath` ou `patchelf` soient installés sur la machine sur laquelle est fait le pack. L'outil `patchelf` est téléchargeable ici: <http://nixos.org/patchelf.html>. Il s'installe facilement à partir des sources. Les distributions Ubuntu fournissent normalement `chrpath`.

Note: pour que toutes les documentations soient incluses dans les packages, il faut avoir les projets **documentation** et **brainvisa-release**, ils font des docs plus ou moins globales sur l'ensemble des projets: docs des commandes en lignes, docs des traitements Axon pour toutes les toolboxes, et docs docbook globales (`cartointernet`, `index.html` dans `doc/`).

Mettre les packs dans `/neurospin/brainvisa/brainvisa_release_candidate` le temps de les tester.

Pack externe (pour internet) :

```
bv_packaging_i2bm -o /neurospin/brainvisa/brainvisa_release_candidate
```

Pack interne (à installer sur `/i2bm/brainvisa` avec un lien pointant dessus dans `/i2bm/local`) :

```
bv_packaging_i2bm -i -o /neurospin/brainvisa/brainvisa_release_candidate
```

Attention : Pour Windows, il n'y a pas d'accès samba sur `/neurospin/brainvisa/brainvisa_release_candidate`, il est donc conseillé d'utiliser `/c/brainvisa_release_candidate`.

Pour chaque release BrainVISA, il faut faire les packs suivants :

- Fedora 4 (externe)
- Mandriva 2008 64 bits (externe)
- CentOS-5.3-x86_64 (interne)
- Ubuntu 64 bits (interne)
- MacOS (externe)
- Windows (interne et externe)

Pour chaque archive, **décompresser et tester** :

```
tar jxvf brainvisa-...
cd brainvisa-...
./BrainVISA
```

Pour les packs internes :

- il faut aussi installer les **modèles SPAM** pour la reconnaissance des sillons. Ils sont téléchargeables sur la page de download de Brainvisa dans `data` :

```
unzip descriptive_models....zip
cp -r models brainvisa-.../share/brainvisa-share-&lt;version&gt;
```

Cela peut aussi se faire via l'interface graphique de Brainvisa en lançant le traitement `Sulci -> morphometry -> Check Spam installation`.

- et la toolbox fMRI (pour les packs linux): faire un nouveau pack de la toolbox ou récupérer celui de la release précédente et l'installer dans le pack Brainvisa.

Pour l'instant, plus de version compatible avec les dernières version de BrainVISA.

```
tar zxvf tbx-fmri...
cp -rf tbx-fmri.../* brainvisa...
```

```
cd brainvisa...<br>
./BrainVISA -b --setup<br>
```

Installation d'un pack stable sur /i2bm/brainvisa

- Copier le pack dans /i2bm/brainvisia :

```
cp -a /neurospin/brainvisa/brainvisa_release_candidate/brainvisa-... <br>/i2bm/brainvisa/Mandriva-2008.0-i686/
```

- Connexion au compte **a-sac-ns-local**
- Modifier le lien /i2bm/local/Mandriva-2008.0-i686/brainvisa pour le faire pointer sur le nouveau pack.

Diffusion des packs externes (pour internet)

- mettre le pack sur le ftp cea : anatomist@ftp.cea.fr, dans binary.
- avoir pris soin d'avoir créé les **versions tags**, et d'avoir régénéré les listes de projets/versions:

```
bv_create_branch release aims anatomist axon brainvisa-share documentation soma-base soma-qtgu  
i soma-io brainvisa-cmake brainvisa-release morphologist-private morphologist-gpl sulci-privat  
e cortical_surface brainrat "old_connectomist*" datamind  
bv_update_projects_list
```

(et copier/coller les lignes des "svn commit")
puis remettre les sources/projets à jour sur le compte a-sac-ns-brainvisa:

```
svn update /neurospin/brainvisa/sources/development/brainvisa-cmake/branches/*  
cd /neurospin/brainvisa/build/Ubuntu-12.04-x86_64/bug_fix  
cmake .  
bv_maker sources
```

- générer les archives des sources avec le script
\$SVN/brainvisa/development/brainvisa-cmake/branches/1.0/bv_tar_source_distribution (passer en argument la liste des projets/composants à packager, eg. opensource). Les archives sont créées dans /tmp. Les copier sur le ftp du cea, dans release.
Cette commande s'appuie sur la présence des sources en versions "tag", donc il faut que l'étape précédente ait été correctement faite.
- mettre à jour le site de brainvisa : annoncer le nouvel update dans news, sur la page d'index, dans la page changelog.
- construire le miroir du site brainvisa et le transférer sur le serveur web :
\$SVN/brainvisa/communication/web/trunk/tools/bv_create_website

Packager un projet seul

```
bv_packaging pack --packname -o <repertoire ou mettre le pack>/<nom du package> --bv_env +project=  
<nom du projet>
```

Par défaut, les dépendances sont packagées, on peut mettre l'option --no-deps pour ne pas packager les dépendances.

On peut utiliser la commande dir au lieu de pack si on ne veut pas compresser le package (par défaut, on obtient un tar.bz2).