

Projet de Programmation

Benoit Donnet
Année Académique 2023 - 2024



1

Agenda

Partie 2: Outils

- Chapitre 1: Compilation
- **Chapitre 2: Librairie**
- Chapitre 3: Tests
- Chapitre 4: Documentation
- Chapitre 5: Débogage
- Chapitre 6: Gestion des Versions

Agenda

- Chapitre 2: Librairie
 - Principe
 - Librairie Statique
 - Librairie Dynamique

Agenda

- Chapitre 2: Librairie
 - Principe
 - ✓ Définition
 - ✓ Compilation
 - Librairie Statique
 - Librairie Dynamique

Définition

- Si on fournit des fonctions rendant un ensemble cohérent de services
 - préférable de grouper les fichiers objets associés sous la forme d'un unique fichier
- Objectifs?
 - facilité de manipulation
 - facilité de mise à jour
- C'est le rôle des fichiers de **bibliothèque**

Définition (2)

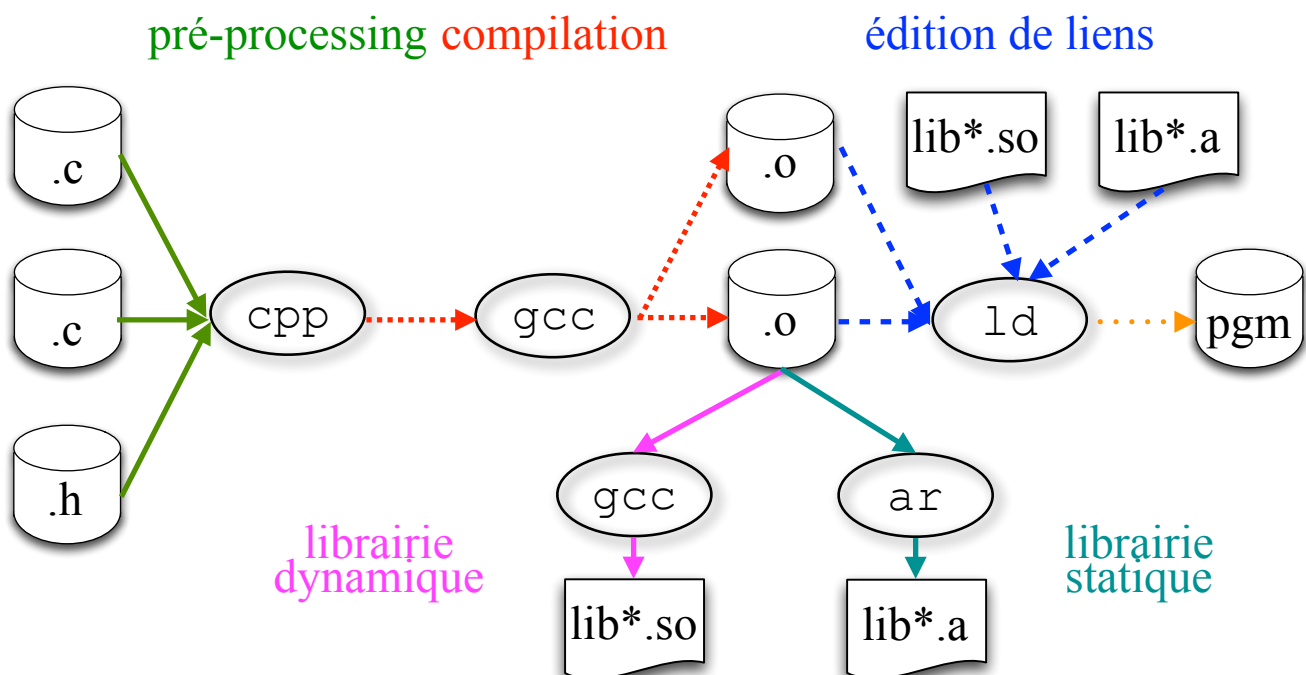
- Fichiers de bibliothèque?
 - fichiers servant à archiver des fichiers "objet"
 - utilisables par l'éditeur de liens
 - ✓ banque de fichiers objet à inclure dans l'exécutable s'ils contiennent la définition de symboles non encore définis

Définition (3)

- Il existe deux types de bibliothèques
 - bibliothèques **statiques**
 - ✓ format?
 - `lib*.a` (Unix)
 - `*.lib` (Dos)
 - ✓ liées à l'exécutable lors de la compilation
 - peut augmenter (fortement) la taille de l'exécutable
 - seul l'exécutable sera encore nécessaire
 - bibliothèques **dynamiques**
 - ✓ format?
 - `lib*.so` (Unix)
 - `*.dll` (Windows)
 - ✓ liées à l'exécutable lors de l'exécution
 - possibilité de mise à jour indépendante des bibliothèques
 - problème si pas présentes (`LD_LIBRARY_PATH`)

Compilation

- Retour sur la compilation multi-fichiers



Agenda

- Chapitre 2: Librairie
 - Principe
 - Librairie Statique
 - ✓ `ar`
 - ✓ `ranlib`
 - ✓ Utilisation
 - Librairie Dynamique

`ar`

- Outil de gestion de fichiers d'archives
 - un fichier archive est un fichier contenant d'autres fichiers
 - peut archiver tout type de fichier
- Il existe d'autres solutions
 - `tar`
 - ✓ plus efficace
 - le format `ar` est reconnu par les éditeurs de liens

ar (2)

- La commande `ar` permet
 - d'ajouter des fichiers à une archive
 - de remplacer des fichiers contenus dans une archive
 - ✓ option `ru` pour remplacer par des plus récents
 - de supprimer des fichiers contenus dans une archive
 - de modifier l'ordre des fichiers de l'archive
 - ✓ l'ordre est important!

```
$>gcc -c brol.c -o brol.o
$>gcc -c brol_io.c -o brol_io.o
$>gcc -c brol_check.c -o brol_check.o
$>gcc -c brol_compute.c -o brol_compute.o
$>ar ruv libbrol.a brol.o brol_io.o brol_check.o
brol_compute.o
```

ar (3)

- Lors de l'édition de liens
 - les fichiers sont parcourus linéairement dans l'archive
 - comportement par défaut
- Quid si 2 fichiers contiennent le même symbole?
 - le 1^{er} fichier rencontré sera inclus
- Quid si, lors de l'édition de liens, le fichier objet a besoin d'un symbole présent dans un fichier déjà vu mais pas inclus?
 - échec de l'édition de liens
- Même problème avec l'ordre dans lequel on demande à l'éditeur de liens de consulter les fichiers bibliothèques

ranlib

- Ajoute dans le fichier d'archives un fichier d'index
 - liste les symboles définis dans les fichiers objet de l'archive
- Objectif?
 - éviter le problème de visibilité dû au parcours linéaire des fichiers de l'archive

```
$>gcc -c brol.c -o brol.o
$>gcc -c brol_io.c -o brol_io.o
$>gcc -c brol_check.c -o brol_check.o
$>gcc -c brol_compute.c -o brol_compute.o
$>ar ruv libbrol.a brol.o brol_io.o brol_check.o
brol_compute.o
$>ranlib libbrol.a
```

ranlib (2)

- Il est intéressant d'ajouter ar et ranlib dans le Makefile

```
#-----
# variables
#-----
AR=ar
RANLIB=ranlib

...

#-----
# Bibliotheque
#-----
libbrol.a : brol.o brol_io.o brol_check.o brol_compute.o
$(AR) ruv $@ $?
$(RANLIB) $@
```

Utilisation

- L'option `-L` du compilateur/éditeur de liens
 - augmenter la liste des répertoires dans lesquels chercher les fichiers de bibliothèque
- L'option `-lxxx` du compilateur
 - permet d'ajouter le fichier `libxxx.a` à la liste des fichiers consultés
 - objectif?
 - ✓ trouver les symboles manquant
 - ✓ dans l'ordre dans lequel ces fichiers sont listés

```
$>cd ../main/  
$>gcc main.c -o main -L../libbrol/ -lbrol -lm
```

Utilisation (2)

- Lorsqu'un symbole non encore résolu se trouve dans un fichier objet d'une bibliothèque, le fichier objet est ajouté à l'exécutable
- Problème
 - un fichier objet contient le code objet de l'ensemble des fonctions présentes dans le code source
 - toutes les fonctions du fichier objet sont donc ajoutées
 - augmentation de la taille de l'exécutable

Utilisation (3)

- Si la taille de l'exécutable est importante
 - il peut être intéressant de répartir les fonctions d'un module en autant de fichiers sources que de groupes de fonctions utilisables séparément

Agenda

- Chapitre 2: Librairie
 - Principe
 - Librairie Statique
 - Librairie Dynamique
 - ✓ Création
 - ✓ Utilisation

Création

- Une librairie dynamique est créée directement à l'aide de gcc
- Deux étapes
 1. création du fichier objet en évitant que le code soit dépendant de l'emplacement mémoire
 2. génération proprement dite de l'archive dynamique

```
$>gcc -fPIC -c brol.c -o brol.o
$>gcc -fPIC -c brol_io.c -o brol_io.o
$>gcc -fPIC -c brol_check.c -o brol_check.o
$>gcc -fPIC brol_compute.c -o brol_compute.o
$>gcc -shared brol.o brol_io.o brol_check.o brol_compute.o
-o libbrol.so
```

Création (2)

- On peut très facilement ajouter la création d'une librairie dynamique au Makefile

```
#-----
# Compilation
#-----
.c .o:
    $(CC) -fPIC -c $(CFLAGS) $(CPPFLAGS) -o $@ $<

#-----
# Bibliotheque
#-----
libbrol.so : brol.o brol_io.o brol_check.o brol_compute.o
    $(CC) -shared -o $@ $?
```

Utilisation

- L'option `-L` du compilateur/éditeur de liens
 - augmenter la liste des répertoires dans lesquels chercher les fichiers de bibliothèque
- L'option `-lxxx` du compilateur
 - permet d'ajouter le fichier `libxxx.so` à la liste des fichiers consultés

```
$>gcc main.c -o main -L../libbrol/ -lbrol
```

Utilisation (2)

- Attention, contrairement aux librairies statiques, les librairies dynamiques sont chargées dynamiquement à chaque exécution
- Donc
 - si la librairie change, l'exécutable change de comportement sans avoir à être recopié
 - ✓ intéressant pour les MAJ, patches, ...
 - si la librairie disparaît/est corrompue/n'est pas trouvée, l'exécutable ne peut être lancé