Technology Tarena 达内科技

Unix系统高级编程

静态库和共享库 Unit02

静态库

Technology Tarena 达内科技

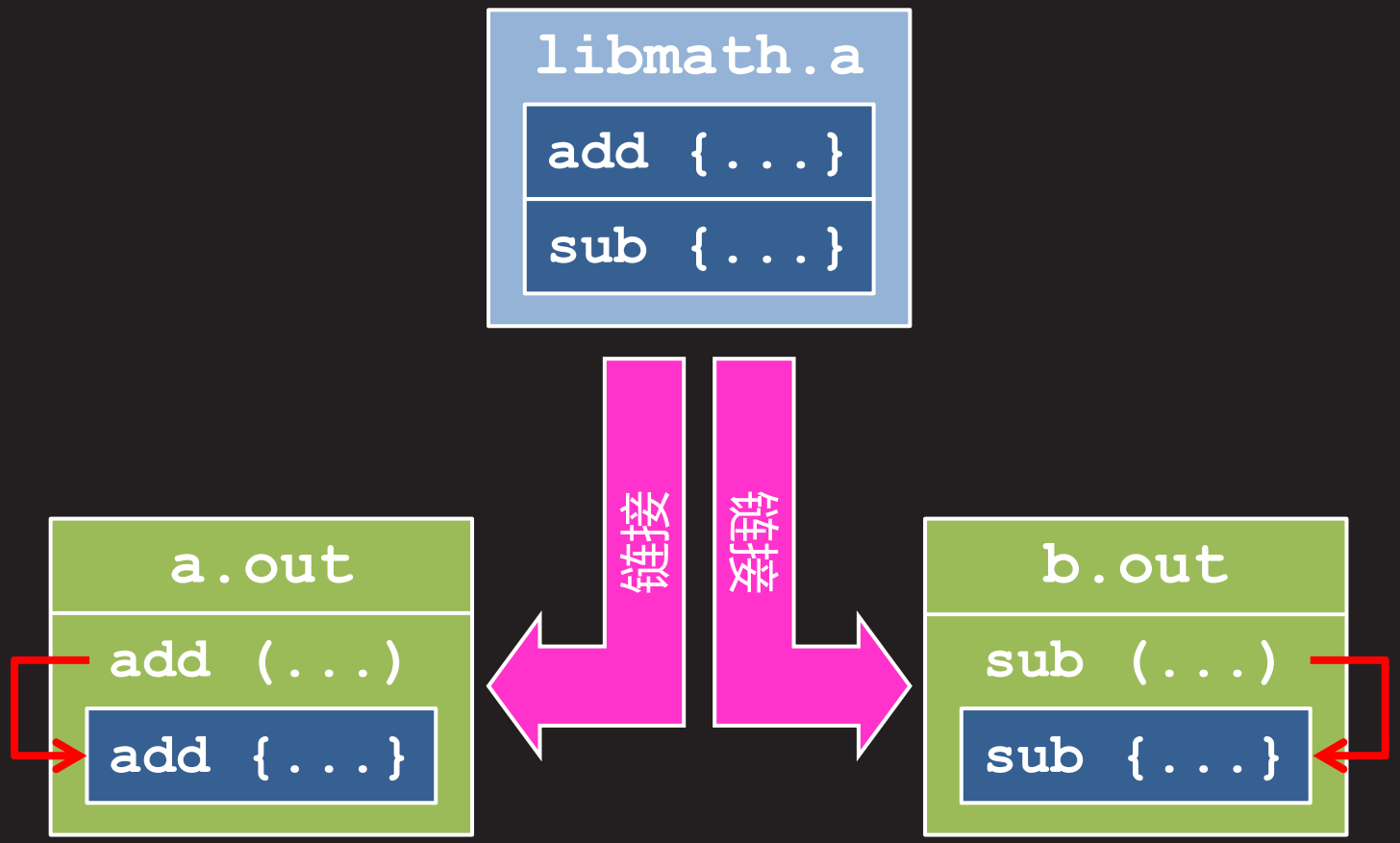
何为静态库

何为静态库

- 为什么要把一个程序分成多个源文件，并由每个源文件编译生成独立的目标文件?
  - 合久必分，化整为零，易于维护
- 为什么要把多个目标文件合并成一个库文件?
  - 分久必合，化零为整，方便使用
- 静态库的本质就是将多个目标文件打包成一个文件
- 链接静态库就是将库中被调用的代码复制到调用模块中
- 静态库占用空间大，库中代码一旦修改必须重新链接
- 使用静态库的代码在运行时无需依赖库，且执行效率高
- 静态库的缺省扩展名是.a

知识讲解

何为静态库（续1）



Technology Tarena 达内科技

构建静态库

构建静态库

- 编辑库的实现代码和接口声明
  - 计算模块：calc.h、calc.c
  - 显示模块：show.h、show.c
  - 接口文件：math.h
- 编译成目标文件

```
$ gcc -c calc.c
$ gcc -c show.c
```

- 打包成静态库文件

```
$ ar -r libmath.a calc.o show.o
```

- 向用户(库的使用者)提供libmath.a和math.h即可

知识讲解

构建静态库（续1）

- ar命令

```
$ ar [选项] <静态库文件> <目标文件列表>
```

-r - 将目标文件插入到静态库中，已存在则更新

-q - 将目标文件追加到静态库尾

-d - 从静态库中删除目标文件

-t - 列表显示静态库中的目标文件

-x - 将静态库展开为目标文件

知识讲解

Technology Tarena 达内科技

使用静态库

使用静态库

- 编辑库的使用代码
  - main.c
- 编译并链接静态库
  - 显示指定库文件的路径

```
$ gcc main.c libmath.a
```

  - 用-l选项指定库名，用-L选项指定库路径

```
$ gcc main.c -lmath -L.
```

  - 用-l选项指定库名，用LIBRARY_PATH环境变量指定库路径

```
$ export LIBRARY_PATH=$LIBRARY_PATH:.
$ gcc main.c -lmath
```

知识讲解

静态库的构建和使用

【参见：static/】

课堂练习

- 静态库的构建和使用

共享库

Technology Tarena 达内科技

何为共享库

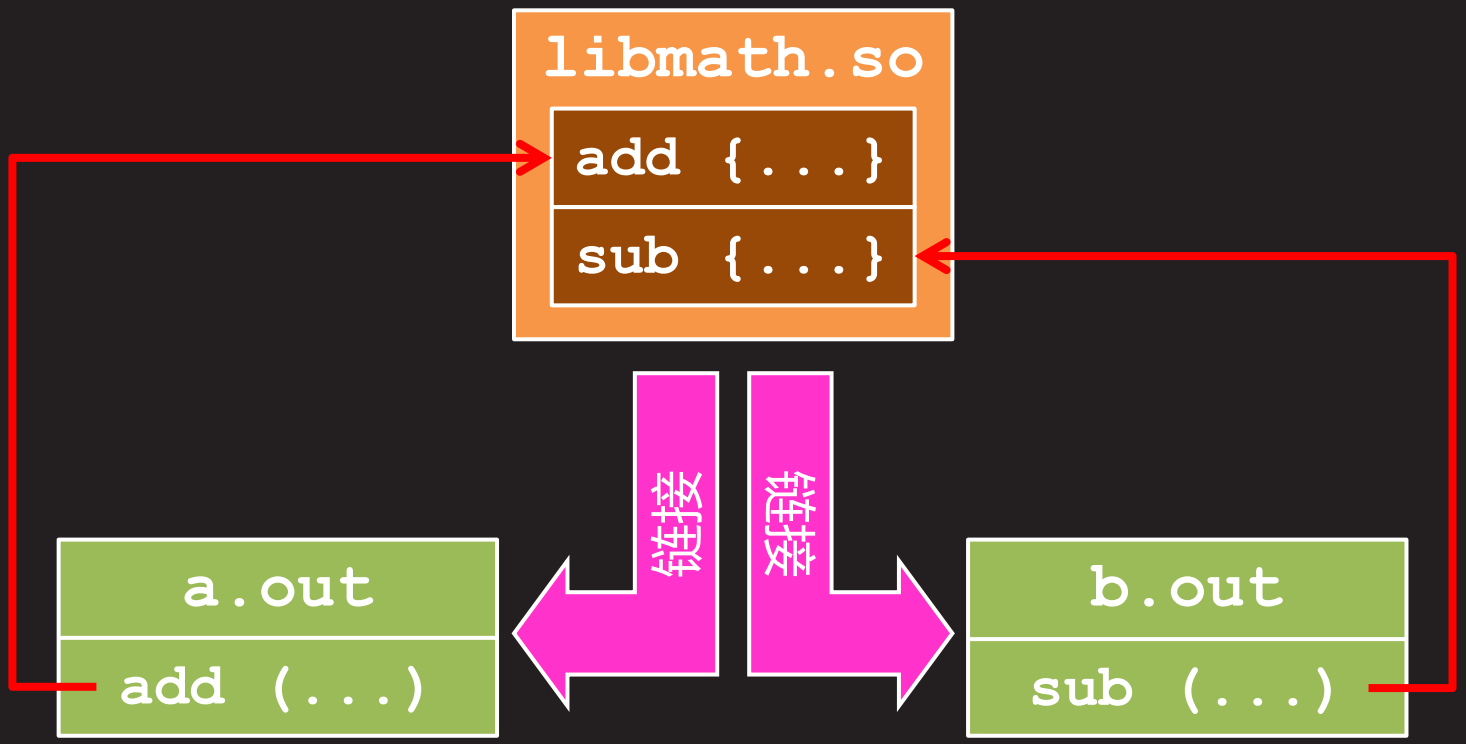
何为共享库

- 共享库和静态库最大的不同就是，链接共享库并不需要将库中被调用的代码复制到调用模块中，相反被嵌入到调用模块中的仅仅是被调用代码在共享库中的相对地址
- 如果共享库中的代码同时为多个进程所用，共享库的实例在整个内存空间中仅需一份，这正是共享的意义所在
- 共享库占用空间小，即使修改了库中的代码，只要接口保持不变，无需重新链接
- 使用共享库的代码在运行时需要依赖库，执行效率略低
- 共享库的缺省扩展名是.so

知识讲解

何为共享库（续1）

知识讲解



Technology Tarena 达内科技

构建共享库

构建共享库

- 编辑库的实现代码和接口声明
  - 计算模块：calc.h、calc.c
  - 显示模块：show.h、show.c
  - 接口文件：math.h
- 编译成目标文件

```
$ gcc -c -fpic calc.c
$ gcc -c -fpic show.c
```

- 连接成共享库文件

```
$ gcc -shared calc.o show.o -o libmath.so
```

- 向用户(库的使用者)提供libmath.so和math.h即可

知识讲解

构建共享库（续1）

- 编译和链接也可以合并为一步完成

```
$ gcc -shared -fpic calc.c show.c -o libmath.so
```

- PIC (Position Independent Code，位置无关代码)
  - 调用代码通过相对地址标识被调用代码的位置，模块中的指令与该模块被加载到内存中的位置无关
  - `-fPIC`：大模式，生成代码比较大，运行速度比较慢，所有平台都支持
  - `-fpic`：小模式，生成代码比较小，运行速度比较快，仅部分平台支持

知识讲解

Technology Tarena 达内科技

使用共享库

使用共享库

- 编辑库的使用代码
  - main.c
- 编译并链接共享库
  - 显示指定库文件的路径

```
$ gcc main.c libmath.so
```

  - 用-l选项指定库名，用-L选项指定库路径

```
$ gcc main.c -lmath -L.
```

  - 用-l选项指定库名，用LIBRARY_PATH环境变量指定库路径

```
$ export LIBRARY_PATH=$LIBRARY_PATH:.
$ gcc main.c -lmath
```


使用共享库（续1）

- 运行时需要保证LD_LIBRARY_PATH环境变量中包含共享库所在的路径

```
$ export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:.
```

- 在可执行程序的链接阶段，并不将所调用函数的二进制代码复制到可执行程序中，而只是将该函数在共享库中的地址嵌入到调用模块中，因此运行时需要依赖共享库
- gcc缺省链接共享库，可通过-static选项强制链接静态库

```
$ gcc -static hello.c
```

知识讲解

共享库的构建和使用

【参见：shared/】

课堂练习

- 共享库的构建和使用

Technology Tarena 达内科技

总结和答疑