



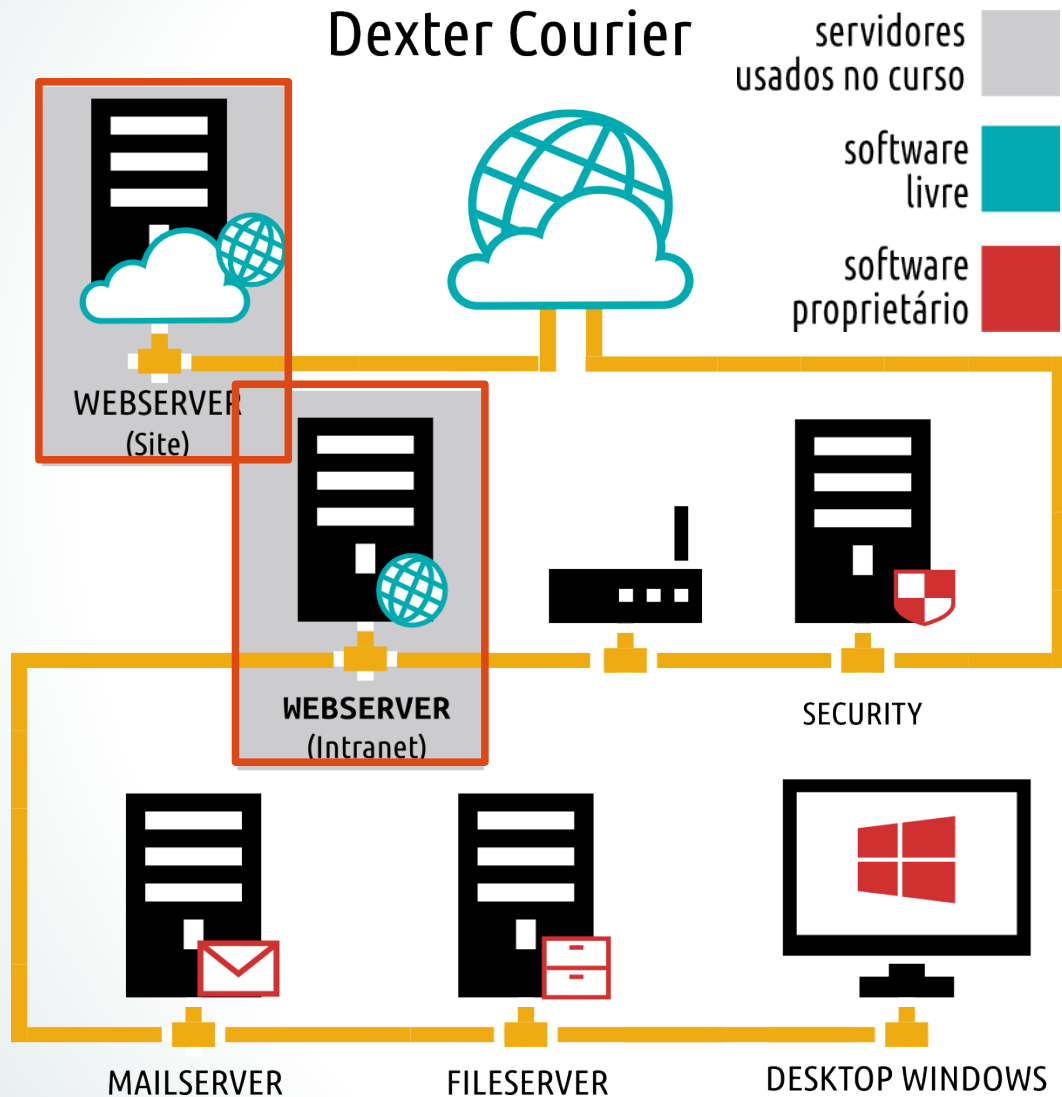
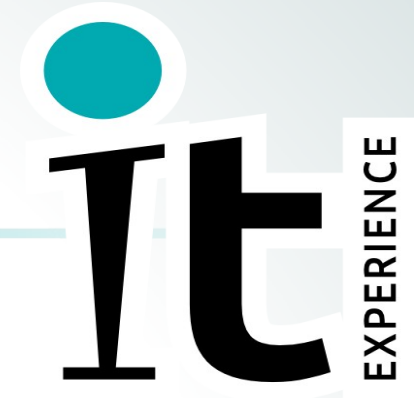
www.4LINUX.com.br

**Só na 4Linux você
aprende
MUITO MAIS!**

Compilação de Programas



IT Experience



Nesta Aula:

➤ Usaremos os Servidores da Dexter:

➤ WebServerInterno

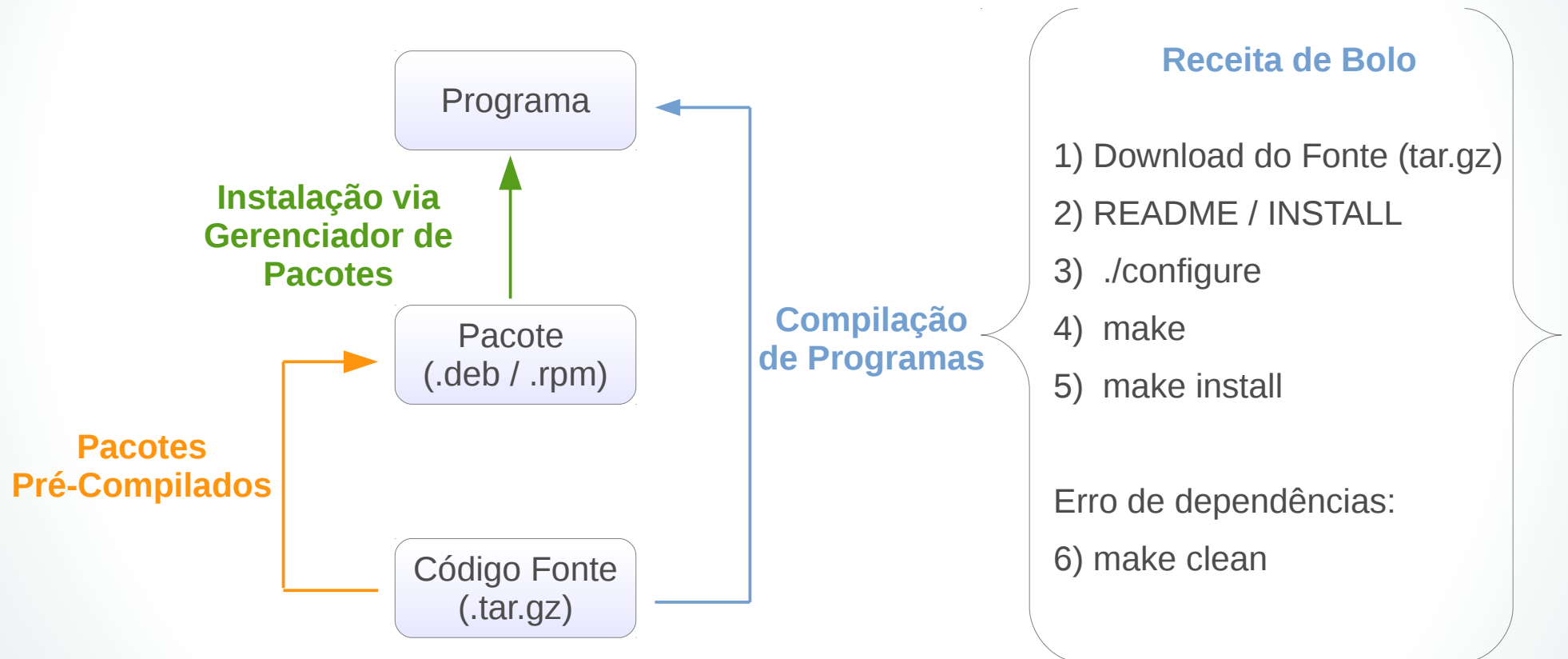


Objetivos da Aula

- Entender os passos de uma Compilação de Programa;
- Instalar o NTOP no WebServerInterno;
- Entender sobre as Bibliotecas Compartilhadas e Estática do Linux.

Compilação de Programas

➤ Métodos de Instalação de Programas do GNU/Linux:



Compilação de Programas

➤ Antes de sair compilando programas em seu servidor, vamos entender os prós e contras dessa técnica: as estáveis.

Prós

- Customização (Performance e Tuning) ;
- Última Versão;
- Processo independente de Distro;

Contra

- Falta de Padronização e Controle;
- Sujeira no Servidor (Pacotes de desenvolvimento);
- Perda de Performance ou má configuração;
- Processo de atualização manual;

Quando Compilar?

- Veja algumas situação de quando você deve optar por compilar um programa:
- A versão pré-compilada da Distribuição não oferece o recurso que você precisa;
- Quando não existe uma versão pré-compilada pela sua Distribuição;
- Quando existe uma necessidade específica de tuning no programa que não é suportado pelo pacote pré-compilado.
- Quando a versão oferecida pela distribuição é muito antiga.

Compilação do Didática

- Vamos compilar um programa criado pela 4Linux para ensinar as etapas de uma compilação;
- As distribuições por padrão não esperam que você compile programas, portanto, não trazem as ferramentas necessárias para se compilar um programa;

```
1# yum install gcc gcc-c++ make
```

gcc → GNU Compiler Collection (GCC) foi desenvolvido originalmente por Richard Stallman;

make → Utilitário de automatização de compilação;

Compilação do Didática

- Vamos compilar um programa criado pela 4Linux para ensinar as etapas de uma compilação;

```
1# wget http://192.168.1.1/450/ola_compilacao.tar.gz
2# tar -xf ola_compilacao.tar.gz -C /usr/local/src
3# cd /usr/local/src/ola_compilacao
4# vim README
5# ./configure
6# make
7# make install
```

```
8# which oi
9# oi
```

Laboratório Dexter



- O NTOP é uma ferramenta de análise de rede, gerando excelentes gráficos de monitoramento das interfaces de rede de uma máquina sobre diversos protocolos;
- Nesse laboratório iremos compilar o NTOP no Servidor WebServerInterno;
- Faça download do Pacote customizado que a 4Linux preparou da última versão do NTOP:

```
1# cd /root/
```

```
2# wget http://192.168.1.1/450/ntop-5.0.1-4linux.tar.gz
```

```
3# ls
```

Laboratório Dexter



➤ Realize a instalação do NTOP conforme INSTALL:

```
1# tar -xf ntop<TAB> -C /usr/local/src
```

```
2# cd /usr/local/src/ntop<TAB>
```

```
3# less INSTALL
```

```
4# ./autogen.sh
```

You must have **libtool** installed to compile ntop.

You must have **automake** installed to compile ntop.

You must have **autoconf** installed to compile autogen.sh.

You must have **wget** installed to compile autogen.sh.

It looks that you don't have the **libpcap** distribution installed.

ERROR: gdbm header or library routines are missing

Error! You need to have zlib.

Dependências....



Laboratório Dexter



Resolvendo Dependências:

1# `yum install libtool automake autoconf wget`

2# `./autogen.sh`

3# `yum install subversion libpcap-devel`

4# `./autogen.sh`

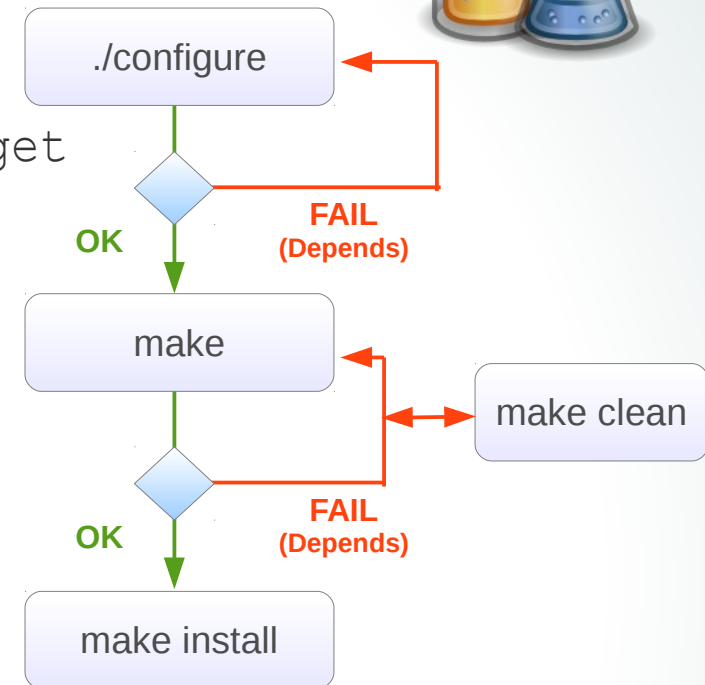
5# `yum install gdbm-devel zlib-devel`

- **Encurtando o Caminho...**

6# `yum install rrdtool-devel python-devel openssl-devel geoip-devel`

7# `./autogen.sh`

Now we're ready to compile ntop (Se as dependências foram satisfeita o ntop está pronto para ser compilado.)

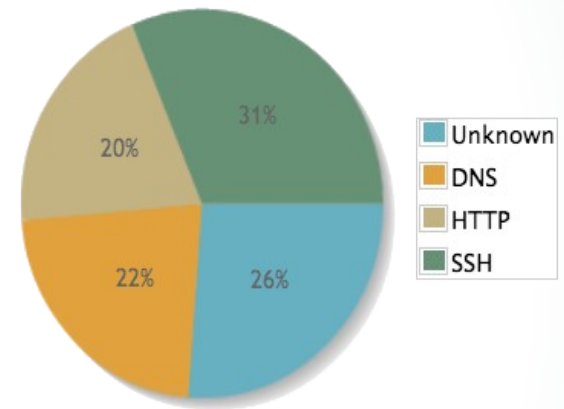


Laboratório Dexter



➤ Configurando o NTOP:

```
1# cd /usr/local/src/ntop<TAB>/confs4linux
2# ./ntop-conf.sh
3# cp ntop.init /etc/init.d/ntop
4# cp ntop.sysconfig /etc/sysconfig/ntop
5# /etc/init.d/ntop start
6# netstat -ntlp | grep 3000
```



Exemplo de Gráfico do Ntop

< Acesse o Browser <http://192.168.200.X:3000> >

Pergunta LPI



Comando usado para instalar uma aplicação através do código fonte compilado através do arquivo Makefile:

- A. make
- B. install
- C. configure
- D. make depend
- E. make install

Pergunta LPI



Comando usado para instalar uma aplicação através do código fonte compilado através do arquivo Makefile:

- A. make
- B. install
- C. configure
- D. make depend
- E. make install

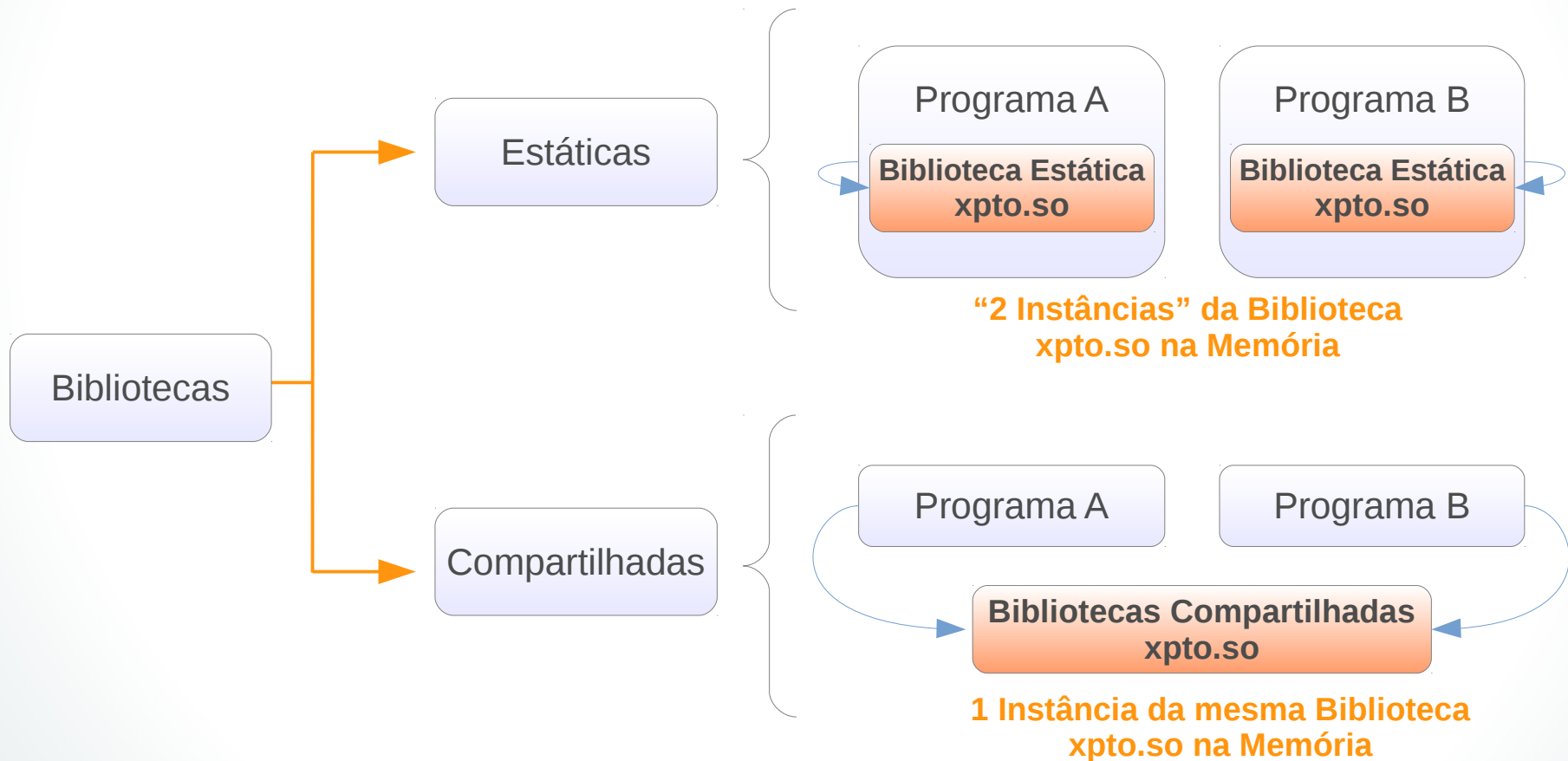
Resposta: E

Bibliotecas no GNU/Linux



Bibliotecas no GNU/Linux

- Existem basicamente 2 tipos de Bibliotecas no GNU/Linux:



Estática ou Compartilhada?

Bibliotecas Estáticas

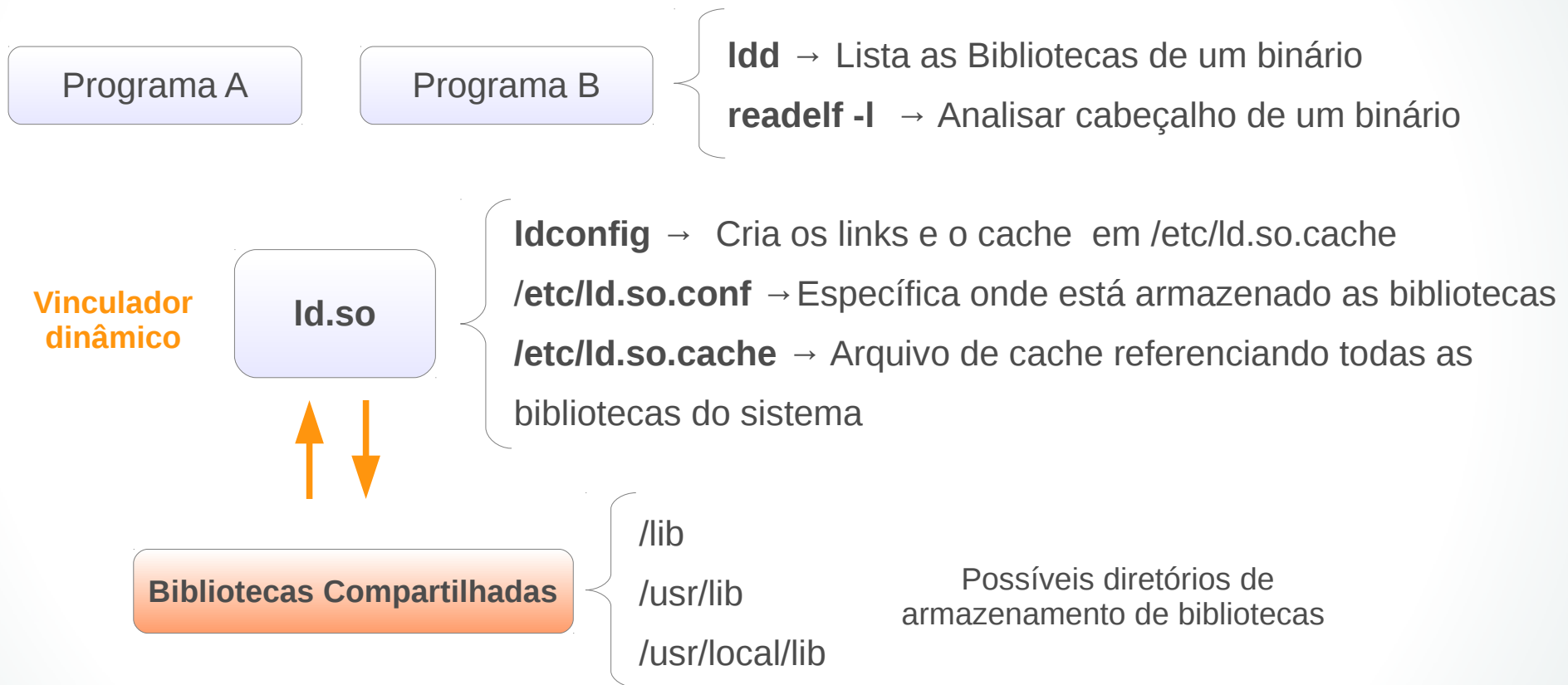
- Binário roda de forma independente; (+)
- Ganho de performance na ótica do aplicativo e não do sistema; (+)
- Gasto desnecessário de memória; (-)
- Tamanho maior no final do arquivo. (-)

Bibliotecas Dinâmicas

- Binário faz referências a várias bibliotecas; (- / +)
- Perda de performance na ótica do aplicativo e não do sistema; (-)
- Gasto menor de memória; (+)
- Tamanho menor final do arquivo. (+)

Bibliotecas Compartilhadas

➤ Funcionamento das Bibliotecas Compartilhadas:



Laboratório Dexter



➤ Praticando Bibliotecas:

```
1# which ntop
2# ldd /usr/local/bin/ntop → Binário com Biblioteca Dinâmica
3# ldd /sbin/ldconfig → Binário com Biblioteca Estática
4# readelf -l /bin/ls
...
[Requesting program interpreter: /lib/ld-linux.so.2]
5# ls -ld /etc/ld*
6# cat /etc/ld.so.conf
7# rm -rf /etc/ld.so.cache
8# ldconfig -p
9# ldconfig
10# ls -ld /etc/ld*
11# ldconfig -p
```

Laboratório Dexter



➤ Simulando problemas de Biblioteca:

```
1# ls /etc/  
2# which ls  
3# ldd /bin/ls | grep libattr  
4# mkdir /opt/lib  
5# mv /lib/libattr.so.1 /opt/lib/  
6# ls /etc/
```

Note que ao movermos uma biblioteca para um diretório desconhecido do do ld.so (Loader) o comando ls parou de funcionar.
Vamos corrigir esse problema!

A biblioteca **libattr.so** é utilizada pela maioria dos comandos no Linux.

DICA: Configurar o novo caminho da biblioteca.

Laboratório Dexter



➤ Simulando problemas de Biblioteca:

```
1# echo "/opt/lib/" >> /etc/ld.so.conf.d/bibliotecas.conf
```

```
2# ldconfig
```

```
3# ldd /bin/ls | grep libattr
```

```
4# ls /etc/
```

A biblioteca **libattr.so** é utilizada pela maioria dos comandos no Linux.

Pronto, agora o comando voltou a funcionar!

Pergunta LPI



Qual o comando irá mostrar as bibliotecas compartilhadas do arquivo `/usr/lib/libpng12.so`?

(Coloque o comando com a biblioteca compartilhada)

Pergunta LPI



Qual o comando irá mostrar as bibliotecas compartilhadas do arquivo `/usr/lib/libpng12.so`?

(Coloque o comando com a biblioteca compartilhada)

Resposta: `ldd /usr/lib/libpng12.so`



www.4LINUX.com.br