

# Utilizando programas através do Apptainer

O Apptainer é um programa para criação e execução de containers, compatível com o Docker e que permite rodar programas dentro de 'imagens' prontas. Isso evita problemas de compatibilidade entre pacotes do sistema e os requeridos pelo programa.

Sua principal vantagem em relação ao Docker é não precisar de permissões especiais dos usuários, tornando-o mais adequado a nossos servidores de uso compartilhado.

Quase tudo que serve para o Singularity serve para o Apptainer.

Abaixo um guia de como rodar um programa dentro do ambiente (no caso, o R).

## Executando uma imagem

As imagens são ambientes com as dependências necessárias para que o programa funcione, por isso é preciso primeiro baixá-las.

[Clique para ver os passos](#)

Para baixar uma imagem podemos usar o subcomando `pull`, como:

```
apptainer pull docker://rocker/ml-verse
```

O exemplo baixa uma imagem do R do [Projeto Rocker](#) que já vem com o R dentro de um ambiente Ubuntu com um repositório de pacotes pré-configurado para instalação mais rápida. A imagem deve ser baixada no diretório onde o comando foi executado, com extensão `.sif` (no exemplo, seria `ml-verse_latest.sif`).

O comando aceita URIs que apontem para imagens que seguem o padrão OCI. Imagens de docker como a do exemplo costumam funcionar.

Com o objetivo de facilitar, **deixaremos algumas imagens pré-baixadas em `/opt/apptainer`.**

Para executar, utilizamos o comando `exec` no escopo da imagem. Como exemplo, utilizaremos a `ml-verse` (já disponível nas máquinas):

```
apptainer exec --nv /opt/apptainer/ml-verse_latest.sif R
```

Ao executar o comando, o apptainer irá criar um container baseado na imagem e executar o programa.

Rodar o Rscript é análogo.

Para rodar o RStudio, siga [o guia](#).

## Particularidades do ambiente

O apptainer irá automaticamente disponibilizar seu \$HOME dentro do ambiente, então seus arquivos devem continuar a ser acessíveis normalmente. A maior diferença se dá nas bibliotecas e outras dependências inclusas na imagem. Caso algum pacote dependa de bibliotecas que não estejam inclusas, o ideal é baixar ou criar uma imagem mais completa.

Para instalar pacotes é preciso configurar o `.libPaths()`. No R, rodar:

```
dir.create(Sys.getenv('R_LIBS_USER'), recursive=TRUE)
.libPaths(Sys.getenv('R_LIBS_USER'))
```

Por padrão isso resultará em pacotes instalados em seu \$HOME na pasta R. É possível alterar a variável de ambiente para mexer nesse comportamento.

## Particularidades dos servidores do IME

Além do \$HOME, o apptainer já disponibiliza o `/tmp` e o `/var/tmp` (entre outros diretórios). Nas brucutus também configuramos acesso ao `/var/fasttmp`, que é sempre um SSD.

## Referências

- <https://apptainer.org/docs/user/latest/>
- <https://rocker-project.org/use/singularity.html>

From:

<https://wiki.ime.usp.br/> - Wiki da Rede IME

Permanent link:

<https://wiki.ime.usp.br/tutoriais:apptainer>

Last update: **2025-05-22 16:03**

