

Compressão e descompressão de arquivos (zip, gzip, rar, tar.gz, tar.bz)

QXD0099 - Desenvolvimento de Software para Persistência

Universidade Federal do Ceará - *Campus* Quixadá

Prof. Francisco Victor da Silva Pinheiro
victorpinheiro@ufc.br



Agenda

- Por que comprimir arquivos?
- Tipos de compressão
- Principais formatos de compressão
- Trabalhando com ZIP
 - Criar um arquivo ZIP
 - Descompactar um arquivo ZIP
- Trabalhando com GZIP e TAR
 - Comprimir com GZIP
 - Descompactar com GZIP
 - Arquivos TAR.GZ
 - Criar e Extrair
- Trabalhando com RAR e TAR.BZ2
 - RAR (necessita software específico)
 - Compactar (WinRAR ou rar no Linux)
 - Descompactar
 - TAR.BZ2
 - Criar e Extrair
- Comprimir e descompactar em Python com bz2

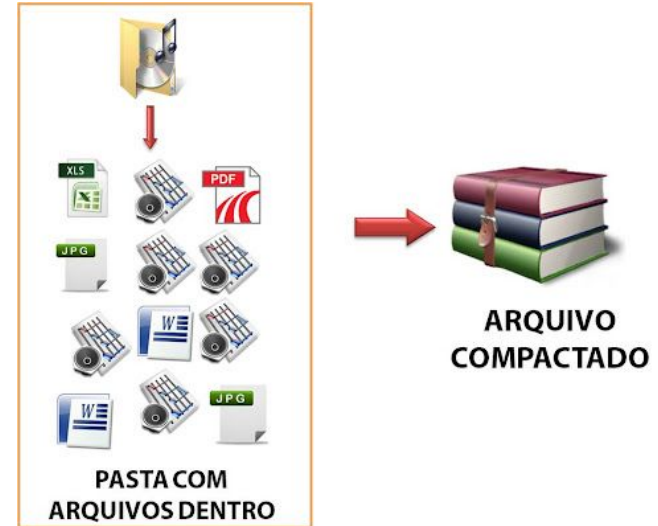
Por que comprimir arquivos?

- Economia de espaço em disco.
- Transferência mais rápida de arquivos.
- Agrupamento de múltiplos arquivos em um só (arquivos de pacotes como TAR).



Tipos de compressão

- **Lossless (sem perda):** Preserva os dados originais (ex.: ZIP, GZIP, TAR.GZ).
- **Lossy (com perda):** Perde detalhes irreversíveis (ex.: JPEG, MP3, MP4).



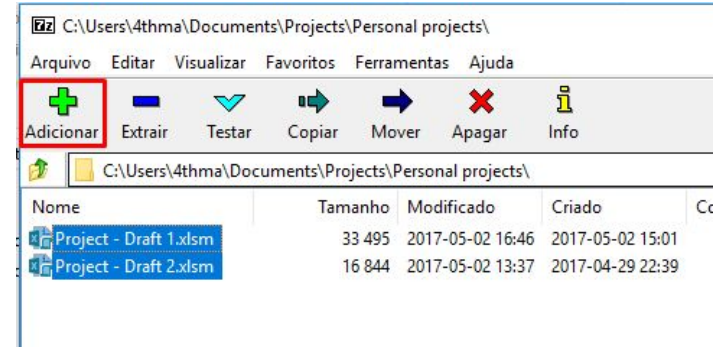
Principais formatos de compressão

- **ZIP:** Amplamente usado, suporta compressão e armazenamento sem compressão.
- **GZIP:** Comprime arquivos individuais.
- **TAR.GZ e TAR.BZ2:** Arquivos TAR comprimidos com GZIP ou BZIP2.
- **RAR:** Proprietário, oferece boa compressão.
- **7z:** Alta compressão, open source.



Trabalhando com ZIP

- **Criar um arquivo ZIP:**
 - **Linux/MacOS:**
 - `zip arquivo.zip arquivo1.txt arquivo2.txt`
 - **Windows:**
 - Botão direito > Enviar para > Pasta compactada.
- **Descompactar um arquivo ZIP:**
 - **Linux/MacOS:**
 - `unzip arquivo.zip`
 - **Windows:**
 - Botão direito > Extrair.



Trabalhando com ZIP

- **Compressão**

```
import zipfile

# Criar um arquivo ZIP
with zipfile.ZipFile('arquivo.zip', 'w') as zipf:
    zipf.write('arquivo1.txt') # Substitua pelo caminho do arquivo
    zipf.write('arquivo2.txt')
```

- **Descompressão**

```
import zipfile

# Extrair um arquivo ZIP
with zipfile.ZipFile('arquivo.zip', 'r') as zipf:
    zipf.extractall('pasta_destino') # Substitua pelo caminho da pasta de destino
```

Trabalhando com GZIP e TAR

- **Comprimir com GZIP:**
 - **Comando Linux/MacOS:**
 - `gzip arquivo.txt`
- **Descompactar com GZIP:**
 - **Comando Linux/MacOS:**
 - `gunzip arquivo.txt.gz`
- **Arquivos TAR.GZ:**
 - **Criar:**
 - `tar -czvf arquivo.tar.gz pasta/`
 - **Extrair:**
 - `tar -xzvf arquivo.tar.gz`

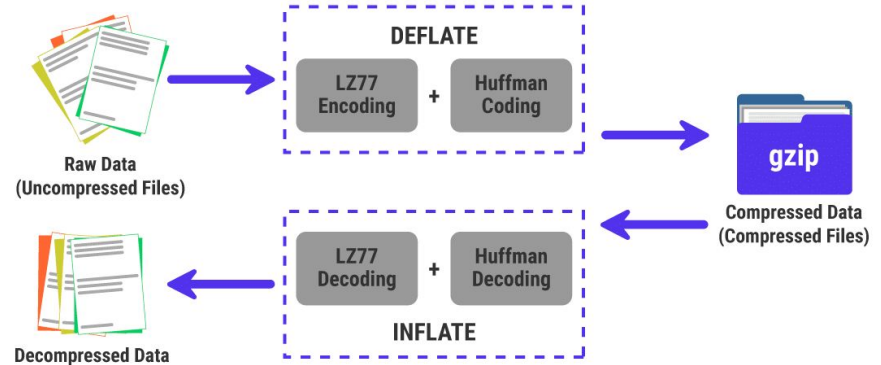
```
import tarfile

# Criar um TAR.GZ
with tarfile.open('arquivo.tar.gz', 'w:gz')
as tar:
    tar.add('pasta/')

# Extrair
with tarfile.open('arquivo.tar.gz', 'r:gz')
as tar:
    tar.extractall('pasta_destino')
```


Trabalhando com RAR e TAR.BZ2

- **RAR (necessita software específico):**
 - **Compactar (WinRAR ou rar no Linux):**
 - `rar a arquivo.rar arquivo1.txt`
`arquivo2.txt`
 - **Descompactar:**
 - `unrar x arquivo.rar`
- **TAR.BZ2:**
 - **Criar:**
 - `tar -cjvf arquivo.tar.bz2 pasta/`
 - **Extrair:**
 - `tar -xjvf arquivo.tar.bz2`



Trabalhando com RAR e TAR.BZ2

- **RAR (necessita software específico):**
 - **Compactar (WinRAR ou rar no Linux):**
 - rar a arquivo.rar arquivo1.txt arquivo2.txt
 - **Descompactar:**
 - unrar x arquivo.rar

```
import os
import rarfile

# Comprimir com RAR (requer instalação do
# programa RAR)
os.system('rar a arquivo.rar arquivo1.txt
arquivo2.txt') # Substitua pelos arquivos

# Descompactar um arquivo RAR
with rarfile.RarFile('arquivo.rar') as rar:
    rar.extractall('pasta_destino') # Substitua
    pelo caminho da pasta de destino
```

Trabalhando com RAR e TAR.BZ2

- **TAR.BZ2:**
 - **Criar:**
 - `tar -cjvf arquivo.tar.bz2 pasta/`
 - **Extrair:**
 - `tar -xjvf arquivo.tar.bz2`

```
import bz2

# Comprimir
with bz2.BZ2File('arquivo.txt.bz2', 'w')
as f:
    f.write(b"Conteudo do arquivo")

# Descompactar
with bz2.BZ2File('arquivo.txt.bz2', 'r')
as f:
    print(f.read())
```

Referências

- <https://docs.python.org/3/library/zipfile.html>
- <https://docs.python.org/3/library/tarfile.html>
- SADALAGE, P. J. E FOWLER, M. NoSQL Essencial. Editora Novatec, São Paulo, 2013.
- REDMOND, E.; WILSON, J. R. Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement. 1ª edição, 2012. The Pragmatic Programmers.
- ULLMAN, J.D.; WIDOW, J. First Course in Database Systems. 3a edição, 2007. Prentice Hall.



Obrigado!

Dúvidas?



Universidade Federal do Ceará - *Campus* Quixadá

Prof. Francisco Victor da Silva Pinheiro
victorpinheiro@ufc.br

