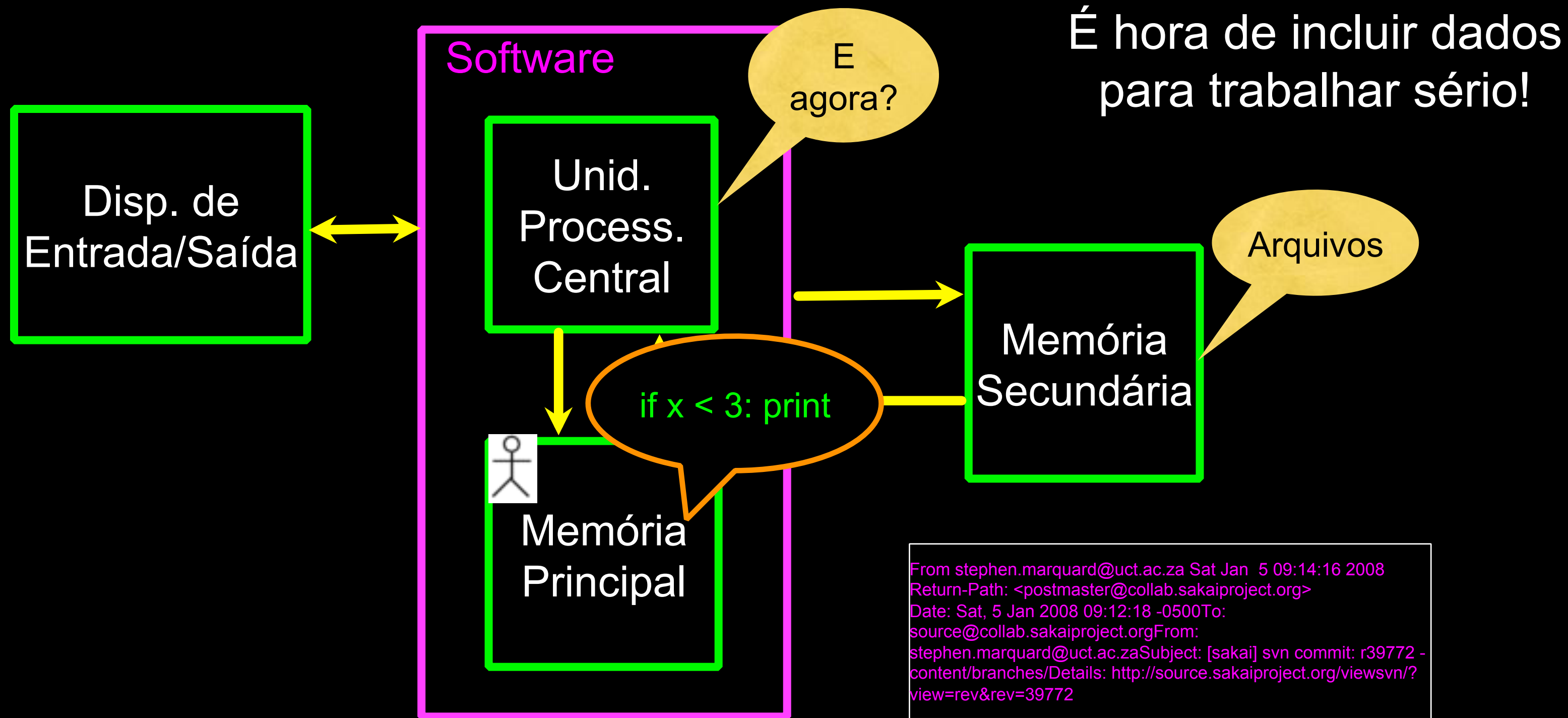


# Manipulação de Arquivos



# Processamento de Arquivos

- Um arquivo de texto pode ser considerado como uma sequência de linhas

From stephen.marquard@uct.ac.za Sat Jan 5 09:14:16 2008

Return-Path: <postmaster@collab.sakaiproject.org>

Date: Sat, 5 Jan 2008 09:12:18 -0500

To: source@collab.sakaiproject.org

From: stephen.marquard@uct.ac.za

Subject: [sakai] svn commit: r39772 - content/branches/

Details: <http://source.sakaiproject.org/viewsvn/?view=rev&rev=39772>

<http://www.py4inf.com/code/mbox-short.txt>

# Abrir um arquivo

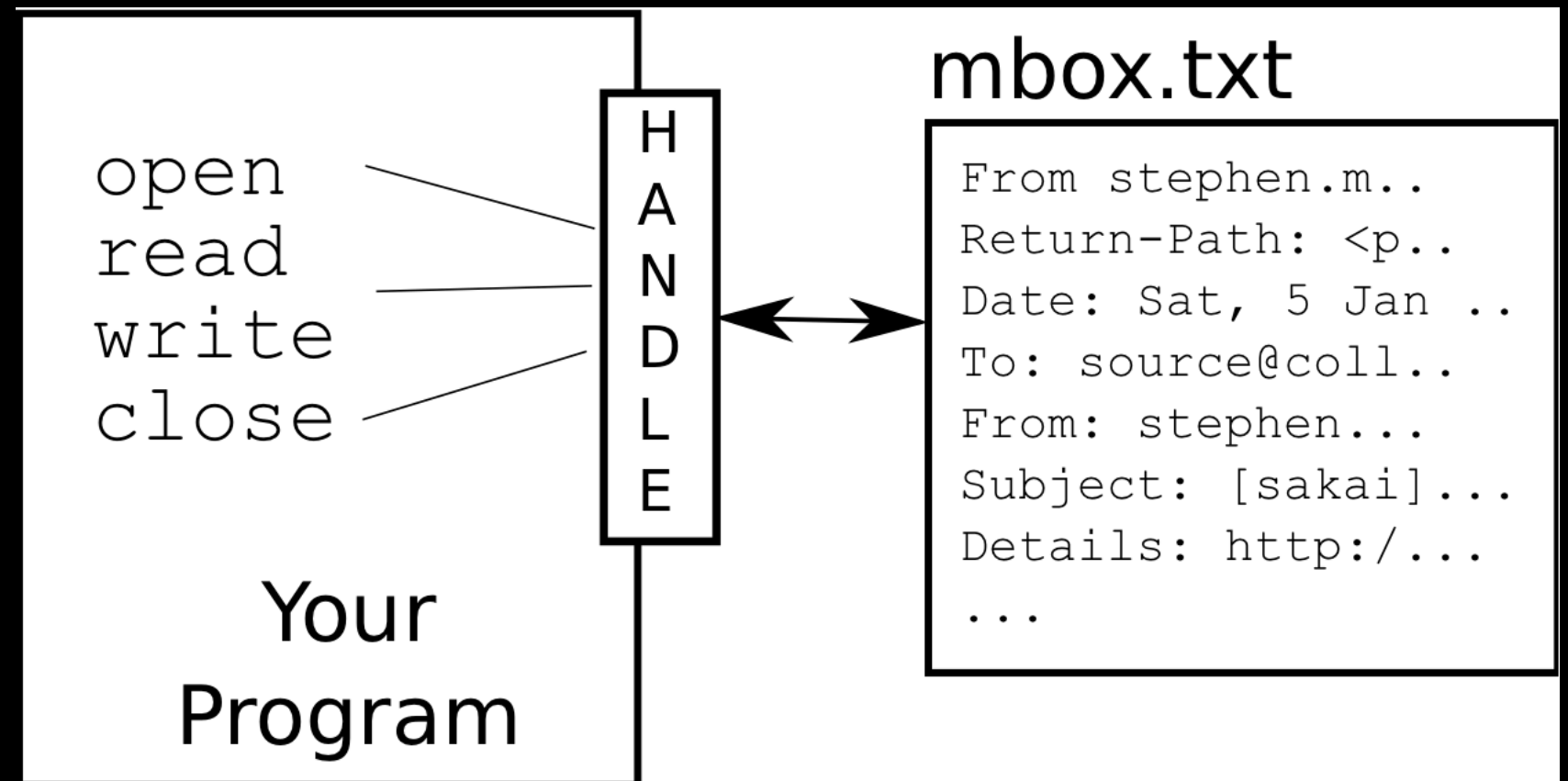
- Para que possamos ler o conteúdo de um arquivo é necessário dizer ao Python qual o arquivo com que pretendemos trabalhar e o que pretendemos fazer
- Esta tarefa é feita recorrendo à função `open()`
- A função `open()` retorna um “`identificador`” - uma variável usada para executar operações no arquivo
- Processo similar a executar “arquivo -> Abrir” num processador de texto

# Usar a função `open()`

- `identificador = open(nome_arq, modo)`
  - `fhand = open('mbox.txt', 'r')`
- retorna um identificador que iremos usar para manipular o arquivo
- `nome_arq` é uma string
- o parâmetro `modo` é opcional; pode ser 'r' no caso de acesso para leitura (modo default), 'w' se quisermos escrever no arquivo (arquivo não precisa existir), 'a' se quisermos adicionar algo ao final do arquivo.

# O que é um identificador?

```
>>> fhand = open('mbox.txt')
>>> print(fhand)
<open file 'mbox.txt', mode 'r' at 0x1005088b0>
```



# Quando os Arquivos não existem

```
>>> fhand = open('stuff.txt')  
Traceback (most recent call last):  File  
"<stdin>", line 1, in <module> IOError: [Errno 2]  
No such file or directory: 'stuff.txt'
```

# O caractere `newline`

- Embora não seja visível, no final de cada linha existe um caractere chamado “`newline`” que indica ao computador o fim de linha
- Este caractere é representado por `\n`
- O caractere `newline` é apenas um e não dois caracteres

```
>>> coisas = 'Ola\nMundo!'
>>> coisas
'Ola\nMundo!'
>>> print(coisas)
Ola
Mundo!
>>> coisas = 'X\nY'
>>> print(coisas)
X
Y
>>> len(coisas)
3
```



# Processamento de Arquivos

- Um arquivo de texto é simplesmente uma sequência de linhas

From `stephen.marquard@uct.ac.za` Sat Jan 5 09:14:16 2008

Return-Path: `<postmaster@collab.sakaiproject.org>`

Date: Sat, 5 Jan 2008 09:12:18 -0500

To: `source@collab.sakaiproject.org`

From: `stephen.marquard@uct.ac.za`

Subject: `[sakai] svn commit: r39772 - content/branches/`

Details: <http://source.sakaiproject.org/viewsvn/?view=rev&rev=39772>

# Processamento de Arquivos

- Um arquivo de texto tem caracteres *newline* no final de cada linha

```
From stephen.marquard@uct.ac.za Sat Jan  5 09:14:16 2008\nReturn-Path: <postmaster@collab.sakaiproject.org>\nDate: Sat, 5 Jan 2008 09:12:18 -0500\nTo: source@collab.sakaiproject.org\nFrom: stephen.marquard@uct.ac.za\nSubject: [sakai] svn commit: r39772 - content/branches/\n\nDetails: http://source.sakaiproject.org/viewsvn/?view=rev&rev=39772\n
```

# Identificador de arquivo como uma Sequência

- Um **identificador de arquivo** aberto para leitura pode ser considerado uma **sequência** de *strings* em que cada linha é uma *string* da sequência
- Podemos então usar a instrução **for** para iterar/percorrer esta **sequência**
- Nota - uma **sequência** é um conjunto ordenado
- A função **close()** fecha o arquivo.

```
xfile = open('mbox.txt')  
for linha in xfile:  
    print(linha)  
xfile.close()
```

# Contar as linhas de um arquivo

- Abrir um **arquivo** em modo de leitura
- Usar um ciclo **for** para ler cada linha
- **Contar** as linhas e apresentar o número de linhas existente

```
fhand = open('mbox.txt')
total = 0
for linha in fhand:
    total = total + 1
print('No. linhas:', total)
```

```
$ python abrir.py
No. linhas: 132045
```

# Ler *\*todo\** o arquivo

- Podemos também ler todo o arquivo (incluindo *newlines*) de uma só vez, colocando-o numa *única string*

```
>>> fhand = open('mbox-short.txt')
>>> texto = fhand.read()
>>> print(len(texto))
94626
>>> print(texto[:20])
From stephen.marquar
```

# Pesquisar num arquivo

- Podemos acrescentar uma instrução **if** no nosso ciclo **for** de forma a apresentar apenas as linhas que cumpram determinado critério

```
fhand = open('mbox-short.txt')
for linha in fhand:
    if linha.startswith('From:'):
        print(linha)
```

# OOPS!

Por que temos linhas em  
branco no resultado?

From: `stephen.marquard@uct.ac.za`

From: `louis@media.berkeley.edu`

From: `zqian@umich.edu`

From: `rjlowe@iupui.edu`

...

# OOPS!

- Por que temos estas linhas em branco?
- Cada **linha** do arquivo tem o caractere ***newline*** no final
- A instrução **print** acrescenta um caractere ***newline*** a cada linha

```
From: stephen.marquard@uct.ac.za\n\nFrom: louis@media.berkeley.edu\n\nFrom: zqian@umich.edu\n\nFrom: rjlowe@iupui.edu\n\n...
```



# Pesquisar num arquivo (corrigido)

- Podemos eliminar os espaços à direita da *string* usando a função `rstrip()` da biblioteca `String`
- O caractere *newline* é considerado um “espaço” e por isso é **eliminado**

```
fhand = open('mbox-short.txt')
for line in fhand:
    line = line.rstrip()
    if line.startswith('From:'):
        print(line)
```

```
From: stephen.marquard@uct.ac.za
From: louis@media.berkeley.edu
From: zqian@umich.edu
From: rjlowe@iupui.edu
....
```

# Saltar com a instrução *continue*

- Podemos saltar uma linha recorrendo à instrução *continue*

```
fhand = open('mbox-short.txt')
for linha in fhand:
    linha = linha.rstrip()
    if not linha.startswith('From:'):
        continue ←
    print(linha)
```

# Usar *in* para selecionar linhas

- Podemos localizar uma *string* numa linha como critério de seleção

```
fhand = open('mbox-short.txt')
for linha in fhand:
    linha = linha.rstrip()
    if '@uct.ac.za' in linha :
        print(linha)
```

```
From stephen.marquard@uct.ac.za Sat Jan  5 09:14:16 2008
X-Authentication-Warning: set sender to stephen.marquard@uct.ac.za using -f
From: stephen.marquard@uct.ac.za
Author: stephen.marquard@uct.ac.za
From david.horwitz@uct.ac.za Fri Jan  4 07:02:32 2008
X-Authentication-Warning: set sender to david.horwitz@uct.ac.za using -f...
```

Perguntando pelo  
nome do arquivo

```
fname = input('Indique o arquivo: ')
fhand = open(fname)
total = 0
for linha in fhand:
    if linha.startswith('Subject:'):
        total = total + 1
print('Existem', total, 'linhas com Subject: no
arquivo', fname)
```

Enter the file name: mbox.txt

Existem 1797 linhas com Subject: no arquivo  
mbox.txt

Enter the file name: mbox-short.txt

Existem 27 linhas com Subject: no arquivo  
mbox-short.txt

# Escrevendo num arquivo

- Podemos usar a função `write()` para escrever em um arquivo.
- O arquivo precisa ser criado/aberto nos modos 'w' ou 'a'.

```
file = open('teste.txt', 'w')  
file.write('Olá\nMundo')  
file.write('Meu nome é Guilherme')
```

Olá  
MundoMeu nome é Guilherme

# Escrevendo num arquivo

- Podemos usar a função `write()` para escrever em um arquivo.
- O arquivo precisa ser criado/aberto nos modos 'w' ou 'a'.

```
fhand = open('mbox-short.txt')
fout = open('mbox-short-out.txt', 'w')
for line in fhand:
    if line.startswith('From:'):
        fout.write(line)
```

```
From: stephen.marquard@uct.ac.za
From: louis@media.berkeley.edu
From: zqian@umich.edu
From: rjlowe@iupui.edu
....
```

# Exercício

Escreva um programa que recebe como entrada o nome de um arquivo e exibe na tela a quantidade de caracteres em cada linha do arquivo lido.

# Exercício

Escreva um programa que recebe como entrada o nome de um arquivo e uma letra, e então gera um novo arquivo (nome diferente) com o mesmo conteúdo do arquivo lido, mas sem qualquer ocorrência da letra informada.



Os direitos de autor (2010) destes slides pertencem a Charles R. Severance ([www.dr-chuck.com](http://www.dr-chuck.com)) da Escola de Informação da Universidade do Michigan, [open.umich.edu](http://open.umich.edu) e são disponibilizados sob Licença Creative Commons Attribution 4.0. Por favor mantenha este último slide em todas as cópias do documento de forma a cumprir com os requisitos de atribuição da licença. Caso faça alguma alteração, pode adicionar o seu nome e organização à lista de contribuições nesta página quando disponibilizar os conteúdos.

Desenvolvimento original: Charles Severance, Escola de Informação da Universidade do Michigan

Contribuições:

Luís Barreto (tradução para Português de Portugal)