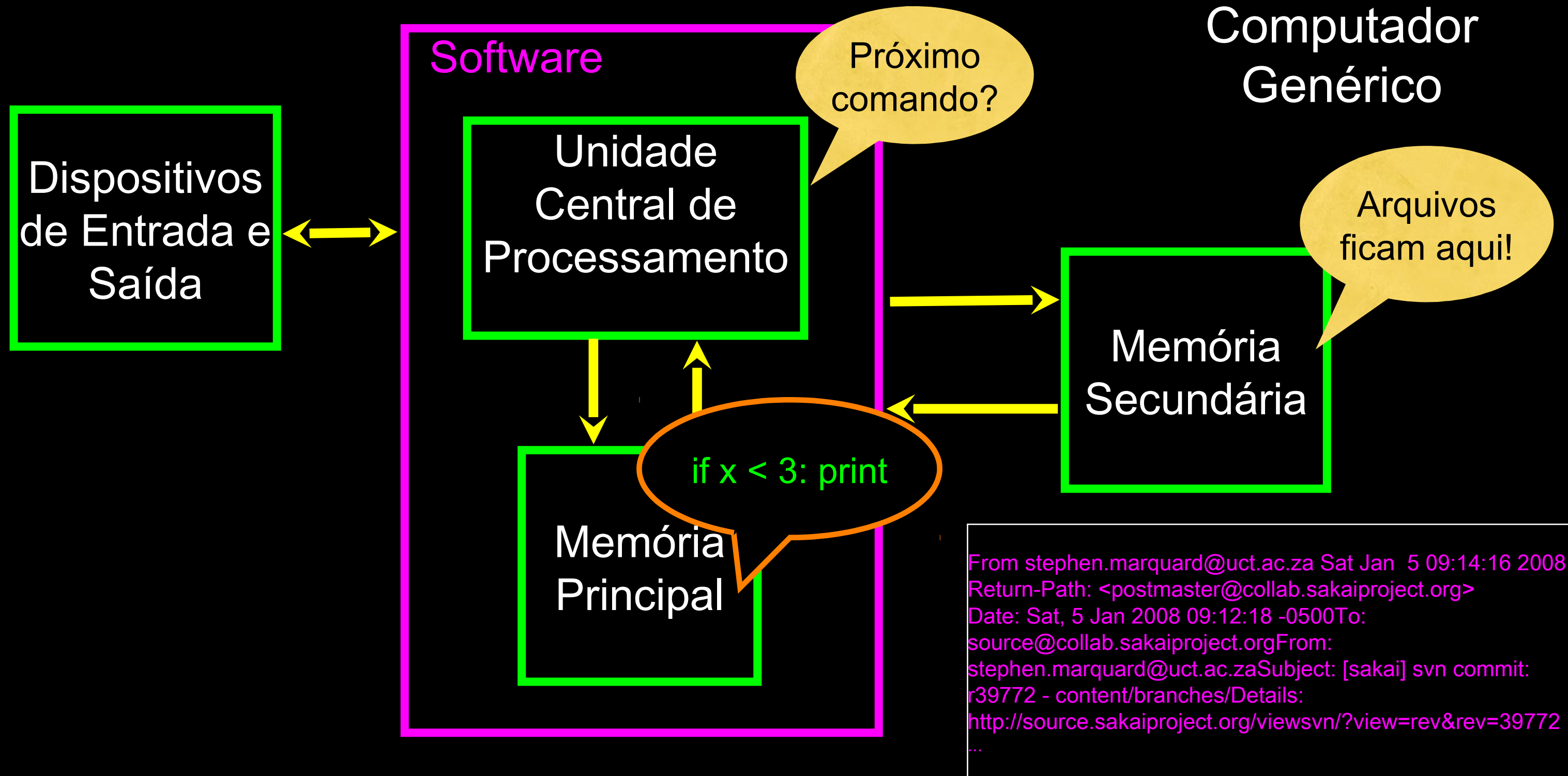


Arquivos

Prof. Alberto Costa Neto
Programação em Python



Processamento de Arquivos

- Podemos imaginar um arquivo como uma seqüência de linhas contendo caracteres

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>

Abrindo um Arquivo

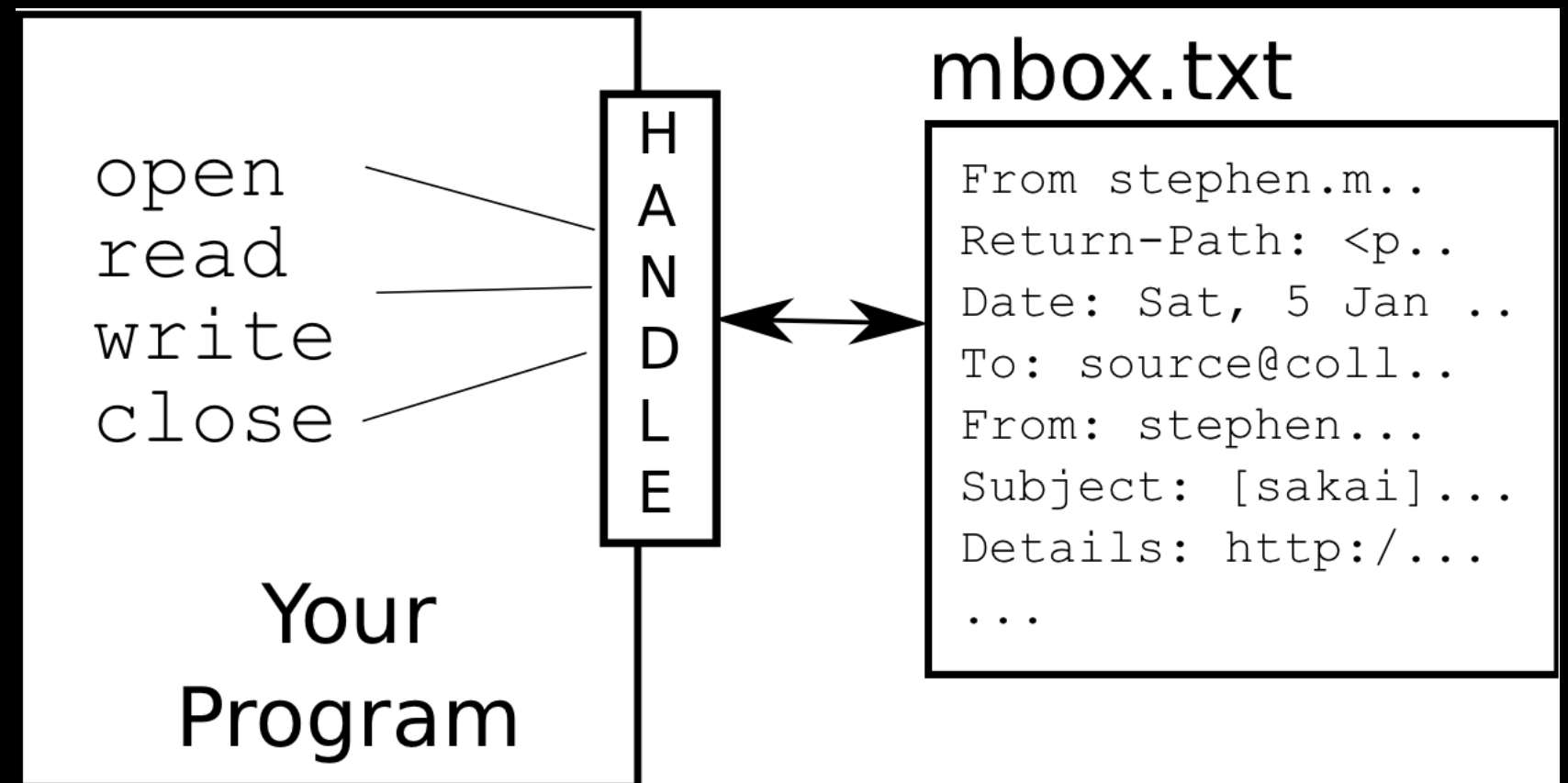
- Antes que possamos ler o conteúdo do arquivo, precisamos informar ao Python com qual arquivo queremos trabalhar e o que desejamos fazer com ele (ler e/ou escrever)
- Isto é feito pela função `open()`
- `open()` retorna um “manipulador de arquivo (handle)” - uma variável que é usada para executar operações em um arquivo
- Similar a acionar “Arquivo -> Abrir” em um Editor de Texto

Usando open()

- `handle = open(nome, modo)` `arq = open('mbox.txt', 'r')`
 - > Retorna um manipulador de arquivo (handle)
 - > `nome` é uma string que identifica o arquivo a ser aberto
 - > `modo` é opcional e deve ser 'r' se o objetivo é ler (read) ou 'w' se pretende-se escrever (write) no arquivo

O que é um manipulador de arquivo (handle)?

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



E quando o arquivo não existe?

```
>>> fhand = open('dados.txt')
```

```
Traceback (most recent call last):  File
```

```
"<stdin>", line 1, in <module>IOError: [Errno 2]
```

```
No such file or directory: 'dados.txt'
```

O caractere `newline`

- Usamos um caractere especial chamado “`newline`” para indicar o final de uma linha
- Ele é representado em uma string com um `\n`
- `newline` é um único caractere (não são 2!)

```
>>> dados = 'Hello\nWorld!'
>>> dados
'Hello\nWorld!'
>>> print dados
Hello
World!
>>> dados = 'X\nY'
>>> print dados
X
Y
>>> len(dados)
3
```


Processamento de Arquivos

- Podemos imaginar um arquivo como uma seqüência de linhas contendo caracteres

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 texto tem caracteres `newline` no final de cada linha

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

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

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

```
To: source@collab.sakaiproject.org\n
```

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

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

```
\n
```

```
Details: http://source.sakaiproject.org/viewsvn/?view=rev&rev=39772\n
```

Handle como uma Seqüência

- Um **handle** de arquivo aberto para leitura pode ser tratado como uma **seqüência** de strings onde cada linha do arquivo é uma string na seqüência
- Podemos usar o comando **for** para iterar na **seqüência**
- Lembre-se que uma **seqüência** é um conjunto ordenado

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

Contando Linhas de um Arquivo

- Abrir o **arquivo** somente para leitura (read-only)
- Usar um laço **for** para ler cada linha
- **Contar** as linhas (incrementando o contador a cada iteração) e imprimir as linhas e imprimir o número de linhas

```
arq = open('mbox.txt')
cont = 0
for linha in arq:
    cont = cont + 1
print 'Linhas:', cont
```

```
$ python contar.py
Linhas: 132045
```

Lendo um Arquivo por completo

- Podemos ler o arquivo completo (com quebra de linhas e tudo) para uma única string

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

OPA!

O que estas linhas em
branco estão fazendo
aqui?

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

From: `louis@media.berkeley.edu`

From: `zqian@umich.edu`

From: `rjlowe@iupui.edu`

`...`

Atenção com a quebra de linha!

O que estas linhas em branco estão fazendo aqui?

- Cada **linha** do arquivo tem um caractere **newline** (**\n**) no final
- O comando **print** acrescenta um caractere **newline** após imprimir o valor

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

Buscando em um Arquivo (corrigido)

- Podemos tirar espaços em branco do lado direito de uma string com a função `rstrip()`
- O caracter `newline (\n)` é considerado “espaço em branco” e é **retirado**

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

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


Pulando iterações com continue


- Durante a leitura de um arquivo, pode ocorrer de termos que ignorar a linha lida, ou seja, não fazer nada com ela.
- O comando **continue** dentro de um comando **if** podem ser usados nesta situação

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

Usando **in** para selecionar linhas


- Podemos buscar uma string em uma linha usando o **in** na **linha** como critério de seleção

```
arq = open('mbox-short.txt')
for linha in arq:
    linha = linha.rstrip()
    if not '@uct.ac.za' in linha :
        continue
    print linha
```

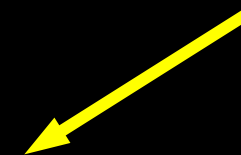


```
From stephen.marquard@uct.ac.za Sat Jan  5 09:14:16 2008
X-Authentication-Warning: set sender to stephen.marquard@uct.ac.za using -f
From: stephen.marquard@uct.ac.za
Author: stephen.marquard@uct.ac.za
From david.horwitz@uct.ac.za Fri Jan  4 07:02:32 2008
X-Authentication-Warning: set sender to david.horwitz@uct.ac.za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

```
nome = raw_input('Digite o nome do arquivo: ')\narc = open(nome)\ncont = 0\nfor linha in arc:\n    if linha.startswith('Subject:') :\n        cont = cont + 1\nprint 'Ha', cont, 'linhas com Subject: em', nome
```



Prompt para
obter o
Nome



Digite o nome do arquivo: mbox.txt
Ha 1797 linhas com Subjec: em mbox.txt

Digite o nome do arquivo: mbox-short.txt
Ha 27 linhas com Subjec: em mbox-short.txt

Arquivos não encontrados

```
nome = raw_input('Nome do arquivo: ')
try:
    arq = open(nome)
except:
    print 'Erro ao abrir o arquivo:', nome
    exit()

cont = 0
for linha in arq:
    if linha.startswith('Subject:') :
        cont = cont + 1
print 'Ha', cont, 'linhas com Subject: em', nome
```

Nome do arquivo: mbox.txt

Ha 1797 linhas com Subject: em mbox.txt

Nome do arquivo: bla bla bla

Erro ao abrir o arquivo: bla bla bla

Buscando em um Arquivo

- Podemos usar um comando `if` dentro de um comando `for` para imprimir apenas as linhas que atendem a um certo critério

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

Gravando um arquivo

- Para gravar um arquivo, basta **abrir no modo escrita** e usar o método **write** quantas vezes for necessário para gravar strings no arquivo

```
arq = open('mbox-short.txt')
arq2 = open('mbox-short2.txt', 'w')
for linha in arq:
    linha = linha.rstrip()
    if not '@uct.ac.za' in linha :
        continue
    arq2.write(linha + '\n')
```

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

X-Authentication-Warning: set sender to david.horwitz@uct.ac.za using -f...



Acknowledgements / Contributions

Agradecimentos / Contribuições



These slides are Copyright 2010- Charles R. Severance (www.dr-chuck.com) of the University of Michigan School of Information and open.umich.edu and made available under a Creative Commons Attribution 4.0 License. Please maintain this last slide in all copies of the document to comply with the attribution requirements of the license. If you make a change, feel free to add your name and organization to the list of contributors on this page as you republish the materials.

Initial Development: Charles Severance, University of Michigan School of Information



These slides were translated and adapted by Alberto Costa Neto (albertocn.sytes.net) of the Federal University of Sergipe