

INTRODUÇÃO

AO

SISTEMA OPERACIONAL

OPEN VMS

Autora:
Daniela Regina Barbetti

Universidade Estadual de Campinas
Centro de Computação
Versão: 4 - Fev/97

Sistema ALPHA-VMS - Introdução

OPEN VMS (Virtual Memory System) é um sistema operacional interativo utilizado pelas máquinas ALPHAs centrais disponibilizadas pelo Centro de Computação. Utiliza DCL (Digital Command Language) para a comunicação entre o usuário e a máquina.

Suas principais características são:

- ⇒ multiusuário
- ⇒ multitarefa (time-sharing)
- ⇒ ambiente interativo/batch
- ⇒ multiprocessamento
- ⇒ processamento de transações
- ⇒ memória virtual
- ⇒ compartilhamento de recursos

NOTAS:

- Multiusuário: permite que vários usuários usem o computador simultaneamente. O sistema operacional gerencia os pedidos que os vários usuários fazem ao computador, evita que um interfira no outro e atribui prioridades quando duas ou mais pessoas querem usar o mesmo arquivo ou a mesma impressora simultaneamente.
- Multitarefa: significa executar mais que uma tarefa ao mesmo tempo, por exemplo, quando você está imprimindo um arquivo e começa a editar um outro documento, você está executando operações multitarefas.
- Multiprocessamento: significa executar vários processos simultaneamente.

Ambiente do Administrador de Sistema

- Manutenção de softwares
- Administração de usuários
- Administração de segurança
- Gerenciamento da performance do sistema
- Contabilidade

NOTAS:

Ambiente do Usuário

- Ambiente interativo
- Processamento batch
- DCL (Digital Command Language)
- Help
- Phone
- Mail
- Editores

NOTAS:

- DCL (Digital Command Language) é um conjunto de instruções em inglês que diz para o sistema operacional realizar operações específicas.

Conectar-se às máquinas ALPHAS

1. Estação de trabalho ou linha discada:

- *Telnet <nome_da_máquina>*

2. Micro na rede da UNICAMP:

- *Telnet <nome_da_máquina>*
- Utilizar software *KERMIT* ou *ST240*

3. Micro com sistema operacional WINDOWS 95

- Utilizar software *WINQVT* ou *QVTNET*

4. Terminal Digital:

- *Connect <máquina>*

5. Terminal IBM:

- *CSFI <máquina>*

NOTAS:

Iniciando

Todo usuário deve ter um *username* e é responsável por uma conta a qual tem uma *password*. Toda vez que o usuário entra na conta, o sistema identifica se o usuário é autorizado ou não e inicia um processo para ele.

A troca da *password* pode ser feita a qualquer momento através do comando *SET PASSWORD*.

Exemplo:

```
$ set pass  
Old password:  
New password:  
Verification:
```

Para sair do sistema basta digitar *LOGOFF*

NOTAS:

- **Abertura de Contas nas máquinas centrais:**

Temos disponível, na máquina TURING, um sistema público de Administração de Contas responsável pela abertura das áreas nas máquinas centrais da UNICAMP, ou seja, máquinas localizadas no Centro de Computação.

Para acessá-lo basta dar o comando "**telnet turing.unicamp.br**" e no *username* digitar "**CONTAS**". A partir desse momento o sistema é auto-explicativo. Para acompanhar o "status" da sua solicitação utilize o módulo de **CONSULTA**. Quando o "status" for "conta bloqueada (senha disponível no Centro de Computação) você deverá dirigir-se ao CCUEC para assinar um termo de compromisso e retirar sua senha inicial.

Comandos

A partir do momento que aparecer o prompt (\$) você poderá digitar um comando DCL, tanto maiúsculo como minúsculo, e teclar <ENTER>. Exemplo;

```
$ show time <ENTER>
```

Sintaxe:

```
[$] [rótulo:] comando [qualificador[=valor]]...] -  
[parâmetro]qualificador...]
```

Exemplo:

```
$ print/copie=5 teste.txt
```

Onde:

\$ é o prompt do DCL

PRINT é o comando DCL

COPIE é o qualificador que modifica o comando

5 é o valor que modifica o qualificador

TESTE.TXT é o parâmetro, no caso o nome do arquivo

NOTAS:

- **Abreviação:**

O VMS permite que os comandos sejam abreviados. Por exemplo, o comando *SET PASSWORD* pode ser abreviado para *SET PASS*, o comando *SHOW* pode ser abreviado para *SH*.

Normalmente abreviamos os comandos para as 4 primeiras letras pois com isso garantimos que não ocorrerá repetição.

Edição de Comandos

◆ Hífen

```
$print/queue=3820/copies=2/del/form=long -  
-$ arquivo_teste.txt
```

◆ Recuperação de um comando

- <CTRL>
- Setas Superior e Inferior
- Comando *RECALL*

◆ Movimentação do cursor

- <CTRL><A>: Muda o modo de digitação (inserção ou sobreposição)
- <CTRL><E>: Fim da linha
- <CTRL><H>: Início da linha
- <CTRL><U>: Apaga linha de comando
- <CTRL><J>: Apaga uma palavra à esquerda do cursor.
- <CTRL><R>: Repete linha de comando

NOTAS:

- O hífen serve para continuar um comando na próxima linha.

Arquivos em disco

- 1 bloco = 512 bytes
2000 blocos = 1 mega
- Todo arquivo reside em um diretório.
- Diretório principal do disco: (MFD) [000000]

- Formato de um arquivo:

Nó::Disco:[Diretório...]nome.extensão;versão

- Formato de um diretório: o diretório é um arquivo com extensão *.DIR* e permite no máximo 8 níveis de subdiretórios.

Sintaxe: [Diretório.Subdiretório...]

- Subdiretório é uma subdivisão do diretório do usuário.
- Quota de disco de WORK

NOTAS:

Editor

◆ Editor: EDT

- Chamando o editor:
`$ED <arquivo>.<extensão>`
- Linha de prompt ou de comando:
*
- Para começar a digitar o texto:
*C
- Incluir um arquivo:
*include <arquivo>.<extensão>
- Sair do modo de edição e voltar ao prompt:
<CTRL><Z>
- Gravar e sair: *exit
- Sair sem gravar: *quit
- Help: *help

NOTAS:

- Todo arquivo possui sua versão e por default esse número é 3.

Comandos mais utilizados

- ◆ DIRECTORY: mostra a lista de arquivos de um diretório e seus atributos.

\$dir/size/date/protection

- ◆ COPY: duplica o conteúdo de um arquivo ou mais arquivos existentes em um novo arquivo.

*\$copy <origem> <destino>/log
\$copy *.*.* [<diretório>]*.*.*
\$copy <origem1>,<origem2> <destino>*

- ◆ SHOW DEFAULT: exibe o diretório corrente.

*\$sh defa
disk1:[curso]*

- ◆ CREATE/DIRECTORY: cria um diretório.

*\$cre/dir [.<subdiretório>]
\$cre/dir [.<subdiretório>.<subdiretório>]*

NOTAS:

- Opções utilizadas na maioria dos comandos:

/log: mostra o resultado da execução do comando.

/confirm: pede a confirmação para o comando ser executado.

/page: mostra o resultado do comando página a página.

- Comando *DIR*:

/size: tamanho do arquivo em blocos.

/date: data/hora da última atualização do arquivo.

/protection: níveis de proteção do arquivo.

Comandos mais utilizados (Cont.)

- ◆ SET DEFAULT: muda o diretório corrente.

```
$set def [.<subdiretório>]  
$set def [-]  
$set def <disco>:[diretório.subdiretório]
```

- ◆ DELETE: apaga um ou mais arquivos.

```
$delete <arquivo>;*  
$delete [.<subdiretório>]<arquivo>;*  
$del/confirm/log
```

- ◆ TYPE: exhibe o conteúdo de um arquivo.

```
$type <arquivo>/page
```

- ◆ SHOW USERS: mostra todas as pessoas que estão usando a máquina.

```
$sh users/full
```

NOTAS:

Comandos mais utilizados (Cont.)

- ◆ **RECALL**: mostra a lista dos últimos 254 comandos utilizados a partir do momento que você entrou na conta.

Listar os comandos disponíveis:

\$recall/all

Apagar o conteúdo do buffer:

\$recall/erase

Reexecutar um comando:

\$recall <no. do comando>

- ◆ **PRINT**: Imprime um ou mais arquivos em uma fila de impressão determinada.

\$print/queue=4090 <arquivo>

NOTAS:

Símbolos

- ◆ Definição: É um nome que representa um número, letra ou valor lógico (como verdadeiro ou falso).
- ◆ Criando um símbolo: para criar um símbolo deve-se utilizar =, ==, := ou :==

Locais:

```
$ss = "show symbol"  
$u := "show users/full"
```

Globais:

```
$dir == "dir/size/date/prot"  
$print:= print/conf/queue=imp1  
$print file.dat
```

- ◆ Verificar o conteúdo de um símbolo:

```
$ show symbol <nome_do_símbolo>
```

NOTAS:

Informações relacionadas à um símbolo:

- Nome: Conjunto de no máximo 255 caracteres composto de letras, dígitos, \$, _, mas não precedido de dígito.
- Valor:
 - Inteiro
 - Cadeia de caracteres
 - Booleanos: TRUE ou FALSE
- Atribuição Local (=): O símbolo é válido no contexto em que está sendo definido e para todos os níveis inferiores.
- Atribuição Global (==): O símbolo é válido em qualquer contexto não importando em que nível foi definido.

OBS: Os símbolos globais \$STATUS, \$SEVETIRY, \$RESTART são reservados para o VMS.

Manipulação de Arquivos

- ◆ APPEND: adiciona o conteúdo de um ou mais arquivos de entrada e coloca em um arquivo de saída já existente.

\$append <origem1>,<origem2> <destino>

- ◆ CREATE: cria um arquivo texto.

\$create <arquivo>.<extensão>

- ◆ DIFFERENCES: compara o conteúdo de dois arquivos e lista a diferença entre os dois.

\$diff <arquivo1> <arquivo2>

- ◆ MERGE: combina dois ou mais arquivos ordenados e grava o resultado em um arquivo de saída.

\$merge <origem1>,<origem2> <destino>

NOTAS:

Manipulação de Arquivos

- ◆ PURGE: elimina as versões antigas de um arquivo.

\$purge <arquivo>/log

- ◆ RENAME: muda o nome de um arquivo.

\$ren <origem> <destino>

- ◆ SEARCH: procura em um arquivo registros que atendam a um determinado critério de casamento de padrão.

*\$search *.*;* <string>*

- ◆ SORT: ordena os registros de um arquivo de entrada e grava ordenadamente em um arquivo de saída.

\$sort <origem> <destino>

NOTAS:

DCL - Tratamento de Erros

Sintaxe: % Utilitário - N - Código, Texto

- Utilitário: Nome do programa que gerou o erro
- **N:** Nível de severidade

S - Sucesso

I - Informativo

W - Aviso

E - Erro

F - Erro fatal ou severo

- Código: Mnemonico que identifica o erro
- Texto: Explicação da mensagem de erro

\$ type me.doc

*%TYPE-W-SEARCHFAIL, error searching for
SYS\$SYSDEVICE:[SHAH]ME.DOC;
-RMS-E-FNF, file not found*

NOTAS:

DCL - Proteção de Arquivos

◆ Tipos de permissão:

- Read: leitura
- Write: escrita
- Execute: execução
- Delete: eliminação

◆ Grupos de proteção:

- System: operador
- Owner: próprio usuário
- Group: usuários do mesmo grupo
- World: todos os usuários

```
$set prot=(s:r,o:re,g,w) <arquivo>  
$sh prot <arquivo>  
$dir/prot
```

NOTAS:

OBS: Deve-se deixar pelo menos a opção de “*read*” para “*system*” para que o operador consiga fazer backups.

MAIL

\$ MAIL

Principais comandos:

- **SEND**: envia uma mensagem.
 - /ed*: chama o editor.
 - /cc*: carbon copy
 - /self*: envia cópia para você mesmo
 - <nome_arquivo>*: envia um arquivo
- **DIR**: mostra a lista das mensagens recebidas.
- **REPLY**: responde uma mensagem.
 - /cc*: carbon copy
 - /ext*: extrai a mensagem e chama o editor.
- **DELETE**: apaga uma ou mais mensagens.
 - /all*: todas as mensagens do folder.
 - del*: mensagem que está sendo lida.
 - del <no_msg>*: apaga a msg especificada.
 - del x,y,z*: apaga as msgs especificadas.
 - del x-z*: apaga as msgs do intervalo.

NOTAS:

Mail é o utilitário da TURING que gerencia suas mensagens recebidas e permite o envio de mensagens com facilidade. Sempre que você tiver recebido uma mensagem, irá receber um aviso ao entrar no sistema.

Ao receber o primeiro mail voce verá que o arquivo **mail.mai** é criado automaticamente. Esse arquivo é protegido e nunca deverá ser apagado, pois trata-se do gerenciador do seu utilitário, bem como os arquivos **mail*.mai**.

Quando você recebe uma mensagem, o folder (pastas) **newmail** é criado automaticamente e uma vez lida, ela passa também automaticamente para o folder **mail**.

Caso você queira saber qual editor está sendo utilizado dentro do MAIL, basta digitar no prompt desse utilitário o comando *sh editor*. Para alterar o editor, basta digitar o comando *set editor <nome_do_editor>*.

Abaixo mostramos alguns exemplos de utilização dos comandos do MAIL:

Comando: **SEND**

Exemplo: MAIL>*SEND*
 To: <endereço_eletrônico>
 Subj: <assunto>

Comando: **DIR**

Exemplo: MAIL>*dir*

Comando: **REPLY**

Exemplo: MAIL>1
 MAIL>*REPLY*
 To: <endereço_eletrônico_de_quem_enviou>
 Subj: RE:<subject_do_mail_recebido>

MAIL (Cont.)

- **READ:** lê uma mensagem.
- **FORWARD:** envia mail recebido para outro usuário.

/cc: carbon copy
- **HELP:** fornece informações sobre o utilitário.
- **EXIT:** sai do MAIL e apaga as mensagens deletadas.
- **COMPRESS:** diminuir o tamanho do arquivo MAIL.MAI.
- **EXTRACT:** extrai um mail para um arquivo.
- **MOVE:** mover mensagens para um folder.
- **SELECT:** permite mudança de folder.

NOTAS:

Comando: **DEL**
Exemplo: MAIL>del 1,5-10

Comando: **READ**
Exemplo: MAIL> read <número da mensagem>

Comando: **FORWARD**
Exemplo: MAIL>forward
To: <usuário_destinatário>

Comando: **COMPRESS**
Exemplo: MAIL>compress
\$ del mail.old;*

Comando: **EXTRACT**
Exemplo: MAIL>extract <nome_arquivo>

Comando: **MOVE**
Exemplo: MAIL>move <nome_do_folder>

Comando: **SELECT**
Exemplo: MAIL>select <nome_do_folder>

Phone

- Chamar usuário: ***\$phone <username>***
- Responder chamada: ***\$phone answer***
- Entrar no utilitário: ***\$phone***

Na linha de comandos:

- ***dial <username>***: chama outro usuário.
- ***answer***: responde a uma chamada.
- ***reject***: rejeita chamadas
- ***dir***: exibe as pessoas que estão utilizando a máquina.
- ***dir <nó>***: exibe as pessoas que estão utilizando a máquina especificada.
- ***help***: fornece informações sobre o utilitário.
- ***exit***: sai do PHONE.

NOTAS:

- Habilitar/Desabilitar ligações:

Para não receber ligações:

\$ SET TERM/NOBROADCAST

Para voltar a receber:

\$ SET TERM/BROADCAST

- Para conseguir parar a tela no comando DIR, utilize a tecla PAUSE/BREAK do micro ou F1 (hold) nos terminais da Digital.

FAQ's

- Endereço eletrônico:

username@TURING.UNICAMP.BR

- Quota em disco:

\$sh quota

- RENPAC

\$sh process/privileges

Aparecerá o identificador psi_domest.

- Utilização da TURING via linha discada

Deve-se abrir uma conta na máquina UNINET-GW.

- Descobrir o username de uma pessoa

\$finger <username >

NOTAS: