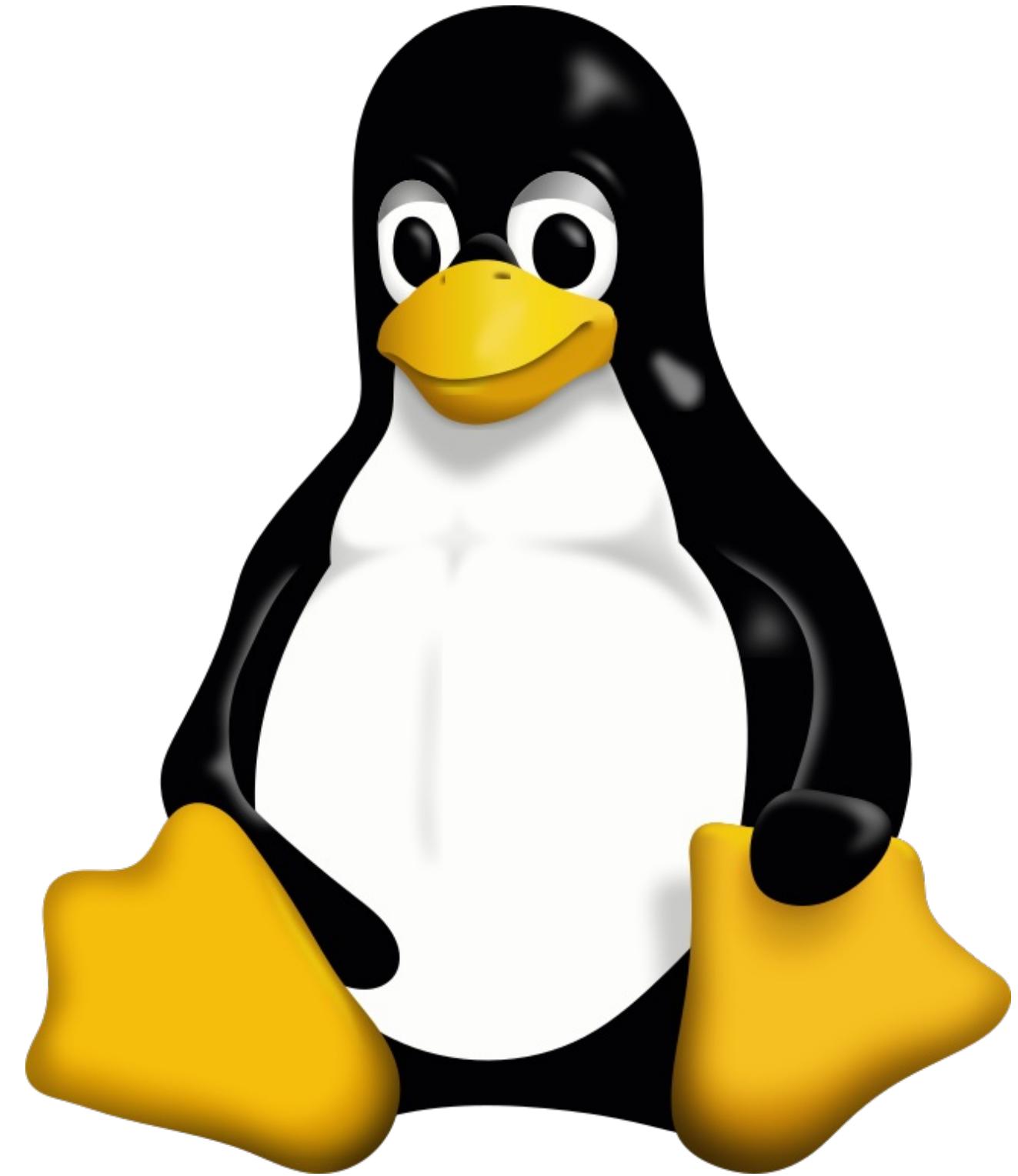
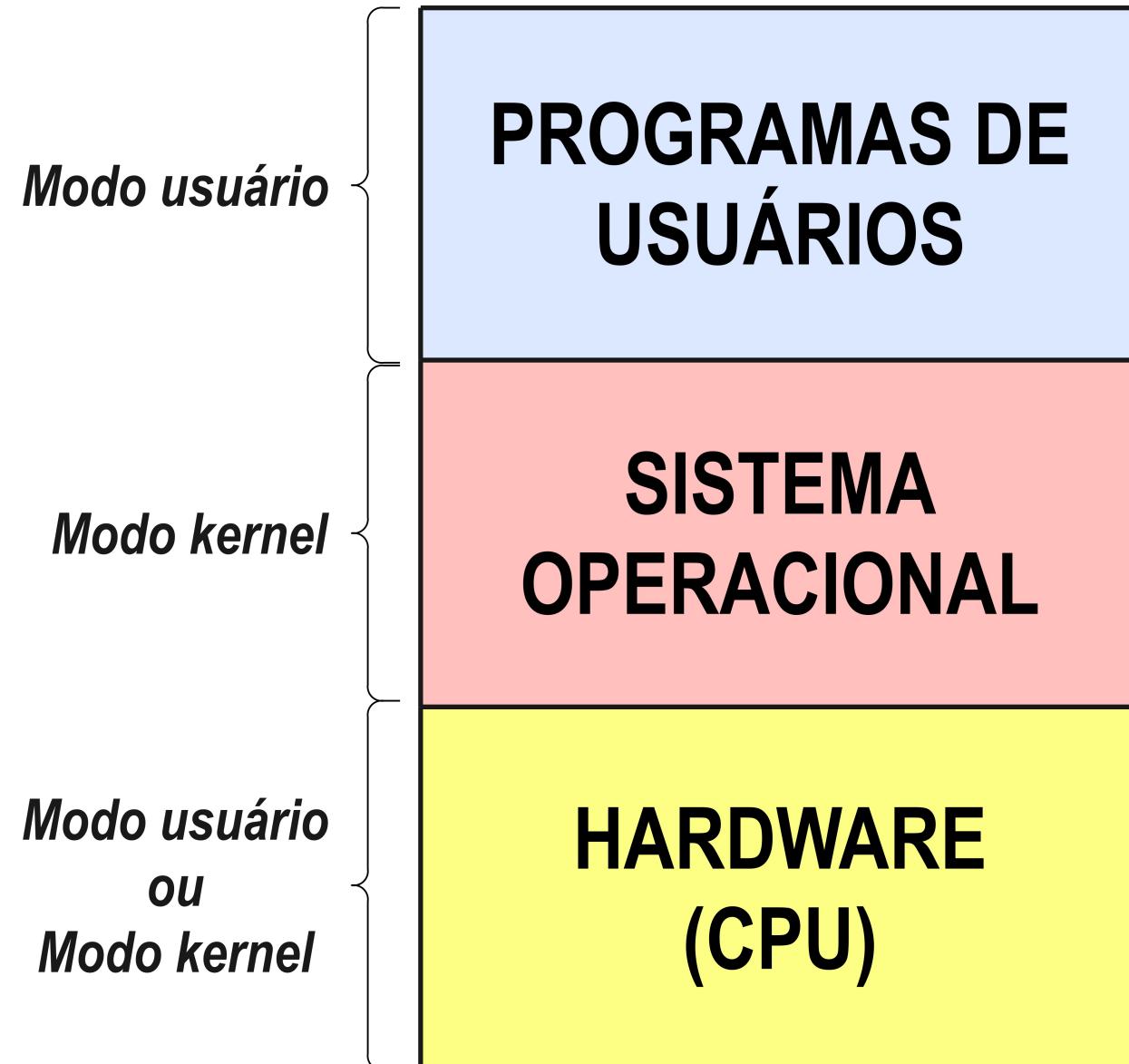


# Sistemas Operacionais: o Shell



lewing@isc.tamu.edu Larry Ewing and The GIMP, CC0,  
Wikimedia Commons (<https://commons.wikimedia.org/wiki/File:Tux.svg>)

# Lembrete: kernel do sistema operacional

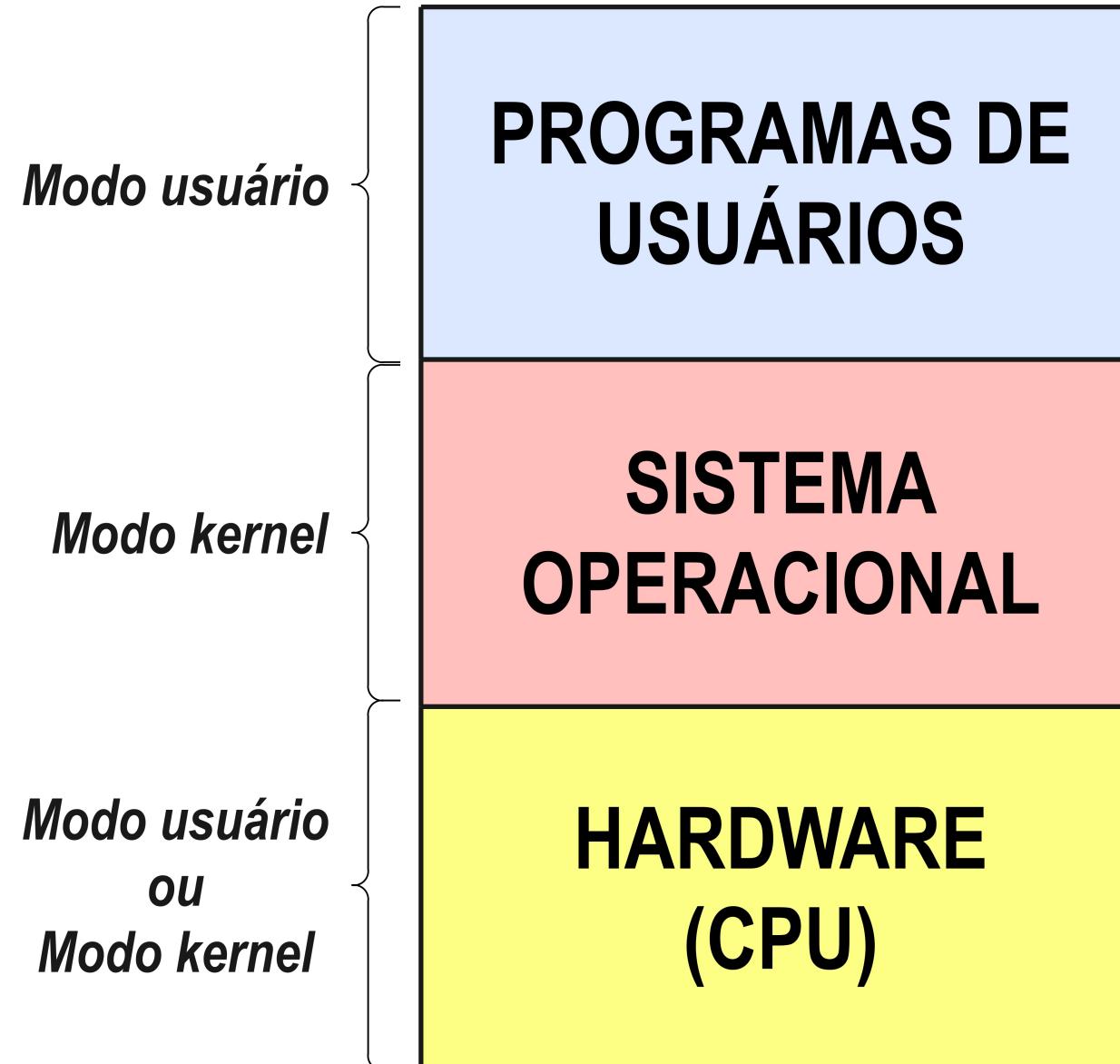


O kernel de um sistema operacional é aquela parte do sistema operacional que está sempre residente em memória (no espaço kernel) e é responsável por 4 grandes funções:

- Gerenciamento de processos
- Gerenciamento de memória
- Gerenciamento de dispositivos
- Chamadas de sistema

Possui controle absoluto sobre todos os aspectos do sistema computacional, atuando como uma ponte entre o software e o hardware.

# Lembrete: modo kernel e modo usuário

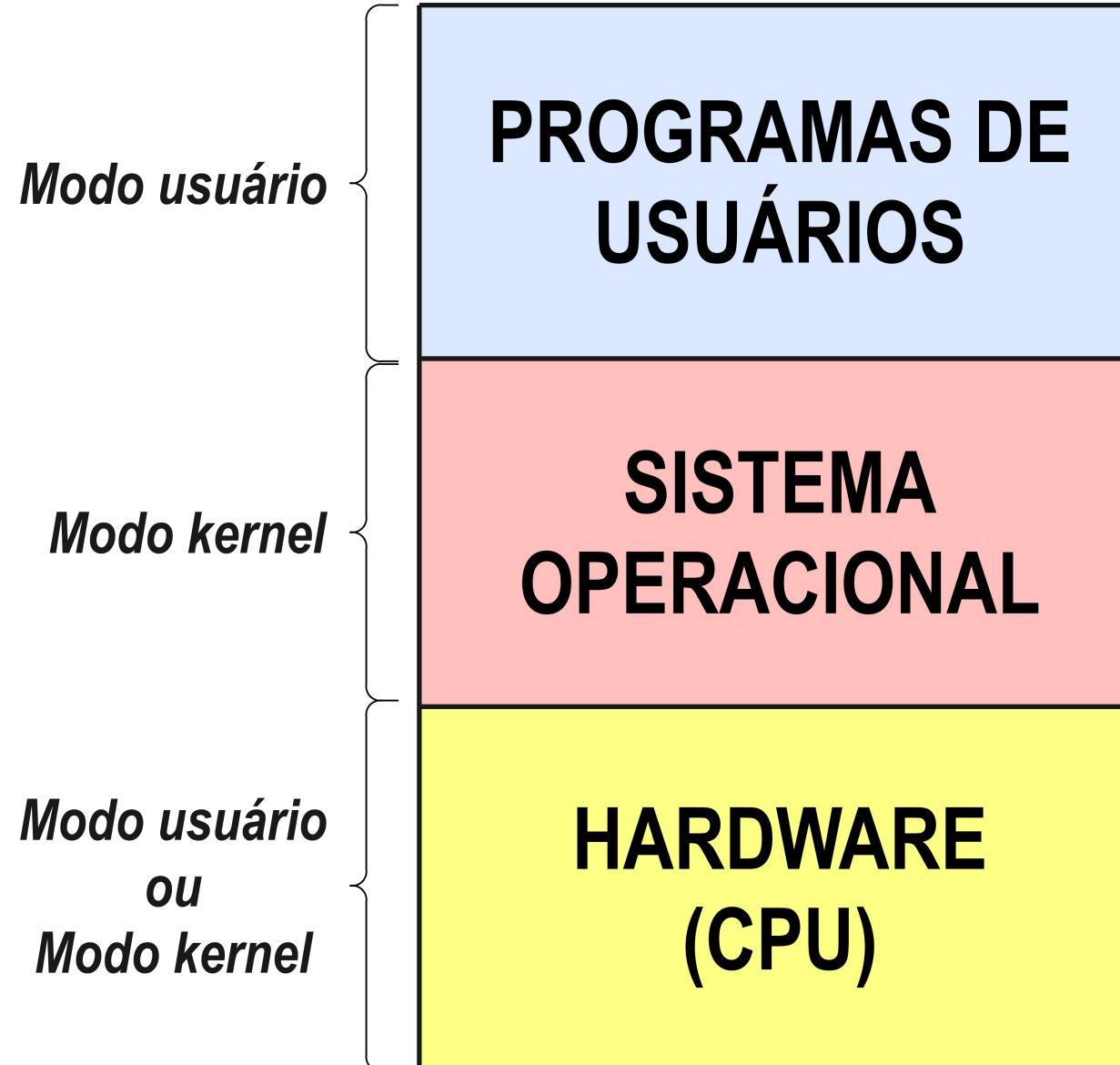
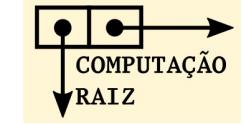


O kernel opera em nível de privilégio elevado, o "**modo kernel**". Nesse nível de privilégio a CPU pode executar qualquer instrução possível, até mesmo as mais "arriscadas" e "inseguras".

Os programas de usuário operam em nível de privilégio mais baixo, o "**modo usuário**". Nesse nível de privilégio a CPU só pode executar um pequeno conjunto de instruções seguras, e o acesso direto à dispositivos do hardware não pode ser feito.

A CPU está sempre operando em um desses dois modos.

# O modo kernel é um mecanismo de proteção!

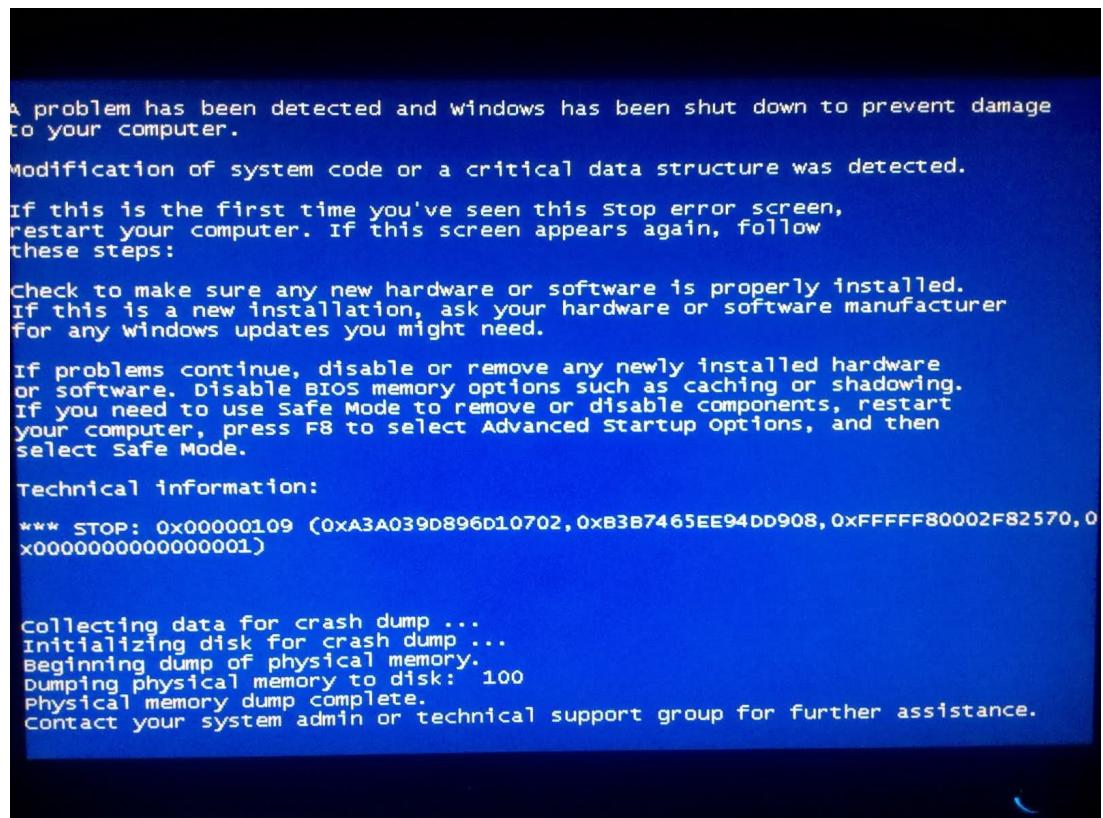


O modo kernel é um mecanismo fundamental de proteção pois **impede que um software aplicativo mal-comportado ou malicioso comprometa todo o sistema**. Imagine o que ocorreria se:

- Um aplicativo mal-programado tentasse sobrescrever áreas de memória críticas?
- Um aplicativo malicioso tentasse apagar dados diretamente dos dispositivos?

Essas são situações que comprometem todo o sistema computacional!

# Se não há proteção... BSOD!



A problem has been detected and windows has been shut down to prevent damage to your computer.

The problem seems to be caused by the following file: SPCMDCON.SYS  
PAGE\_FAULT\_IN\_NONPAGED\_AREA

If this is the first time you've seen this stop error screen, restart your computer. If this screen appears again, follow these steps:

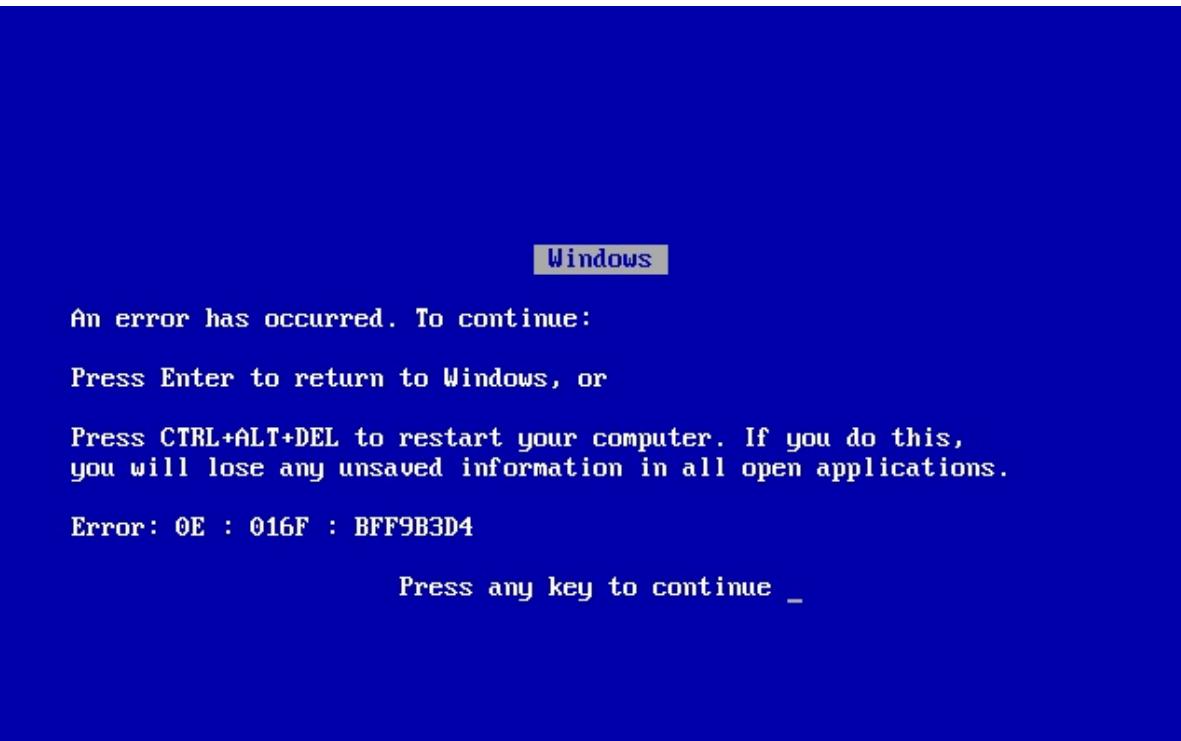
Check to make sure any new hardware or software is properly installed. If this is a new installation, ask your hardware or software manufacturer for any windows updates you might need.

If problems continue, disable or remove any newly installed hardware or software. Disable BIOS memory options such as caching or shadowing. If you need to use safe Mode to remove or disable components, restart your computer, press F8 to select Advanced Startup options, and then select Safe Mode.

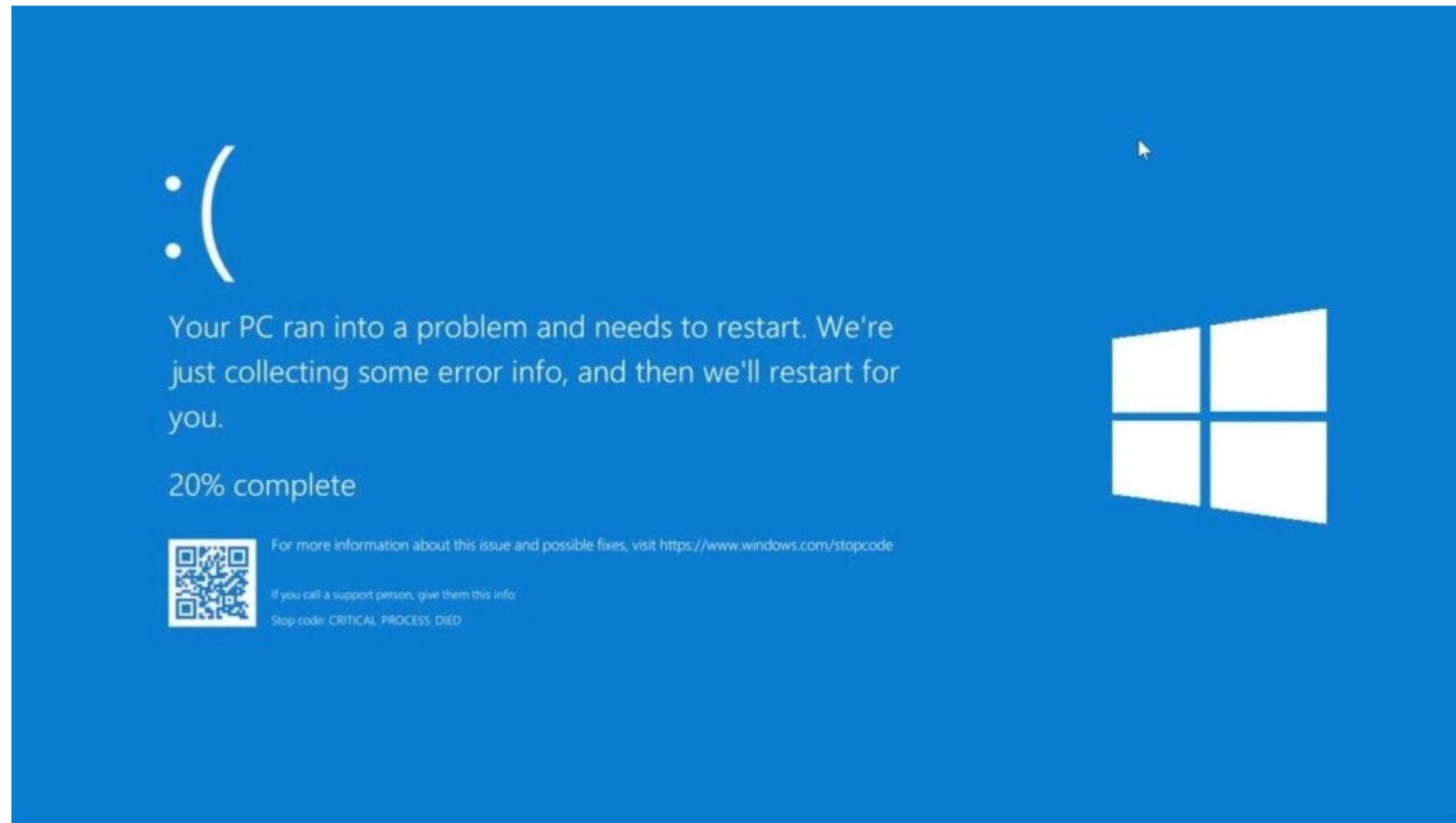
Technical information:

\*\*\* STOP: 0x00000050 (0xFD3094C2, 0x00000001, 0xFBFE7617, 0x00000000)

\*\*\* SPCMDCON.SYS - Address FBFE7617 base at FBFE5000, DateStamp 3d6dd67c



# Se não há proteção... BSOD!



Windows NT crashed.  
I am the Blue Screen of Death.  
No one hears your screams.

Windows caiu.  
A morte azul me envolveu.  
Ninguém vai me ouvir.

Ecoa um suspiro.  
O azul cala a esperança.  
Tudo se perdeu.

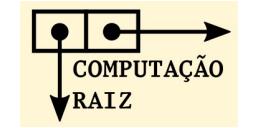
A luz se apagou.  
Azul, a morte vigia.  
Teus gritos... em vão.

Cliquei sem pensar.  
Planilha? Nunca mais vi.  
Obrigado, Bill.

Horas de esforço.  
Um azul e tudo se foi.  
Salvar? Esqueci.

# Se não há proteção... BSOD!

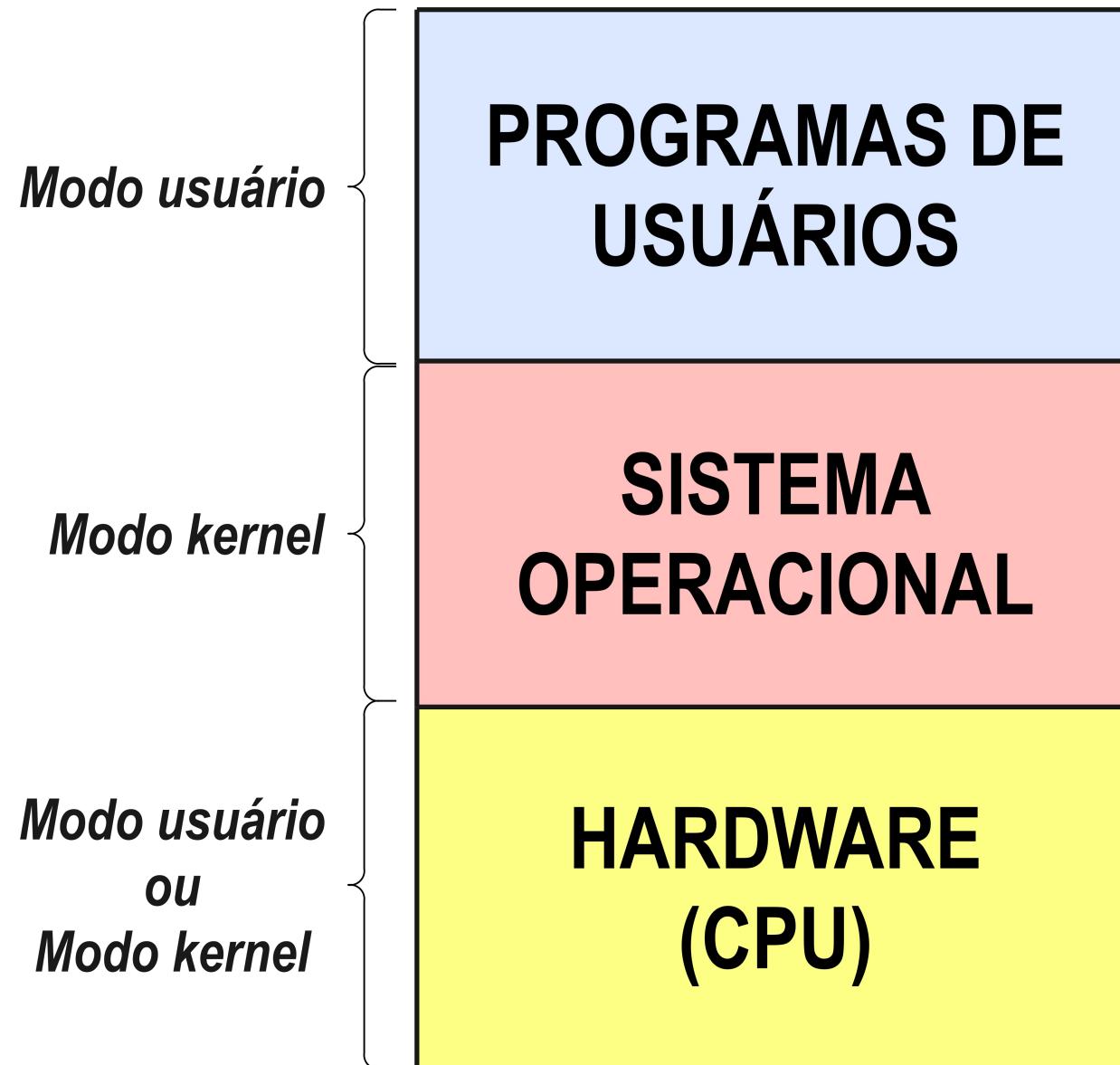
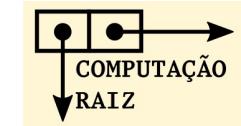
---



---

<https://www.youtube.com/watch?v=XVY44QYsxTgs>

# O mecanismo de proteção também cria um problema

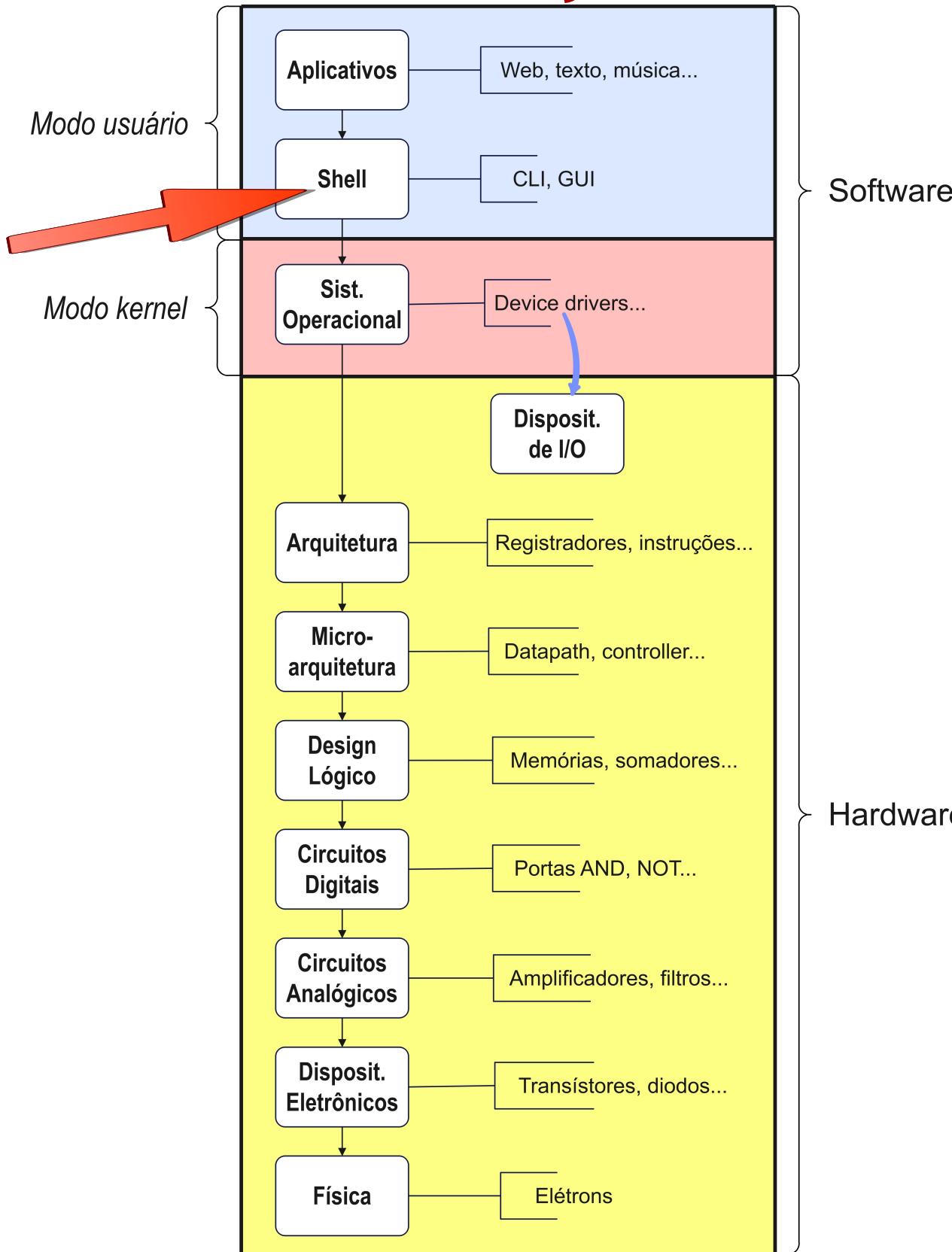


Como o modo kernel é uma proteção, se um aplicativo de usuário falhar o kernel geralmente consegue conter a programa e encerrá-lo sem que o sistema operacional inteiro pare de funcionar (**BSOD**).

Mas essa barreira de proteção também cria um problema? **como um programa em modo usuário pode usar serviços que apenas o kernel pode fornecer, como ler um arquivo do disco ou enviar um arquivo pela rede?**

Qual a solução?

# O SHELL é a solução!

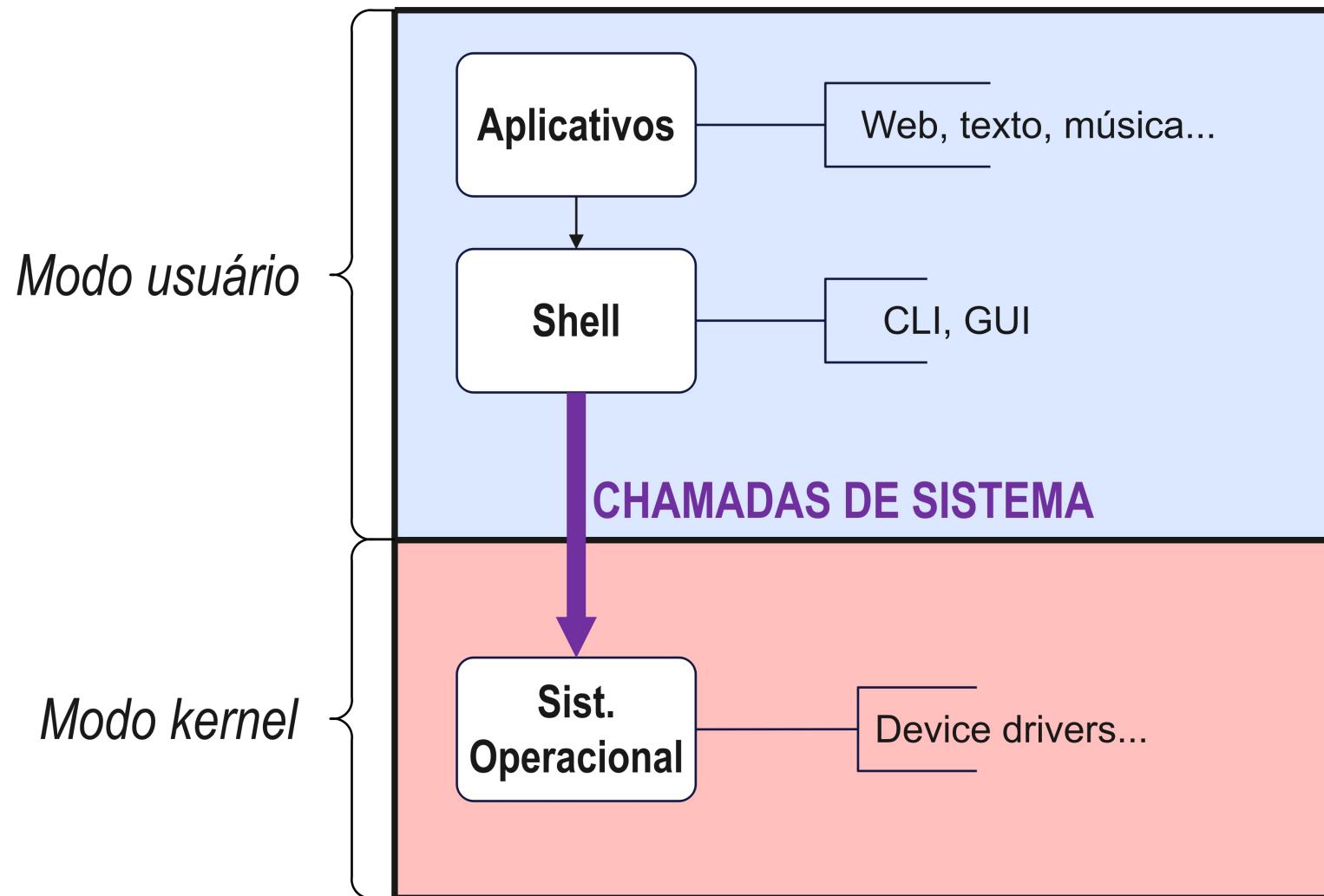


**O shell é a interface para o sistema operacional, a ponte entre os programas do usuário e o kernel.**

**A função principal de um shell é fornecer uma maneira para que os usuários e programas do usuário possam acessar os serviços do sistema operacional de forma controlada e segura.**

**O shell faz isso através das chamadas de sistema (system calls).**

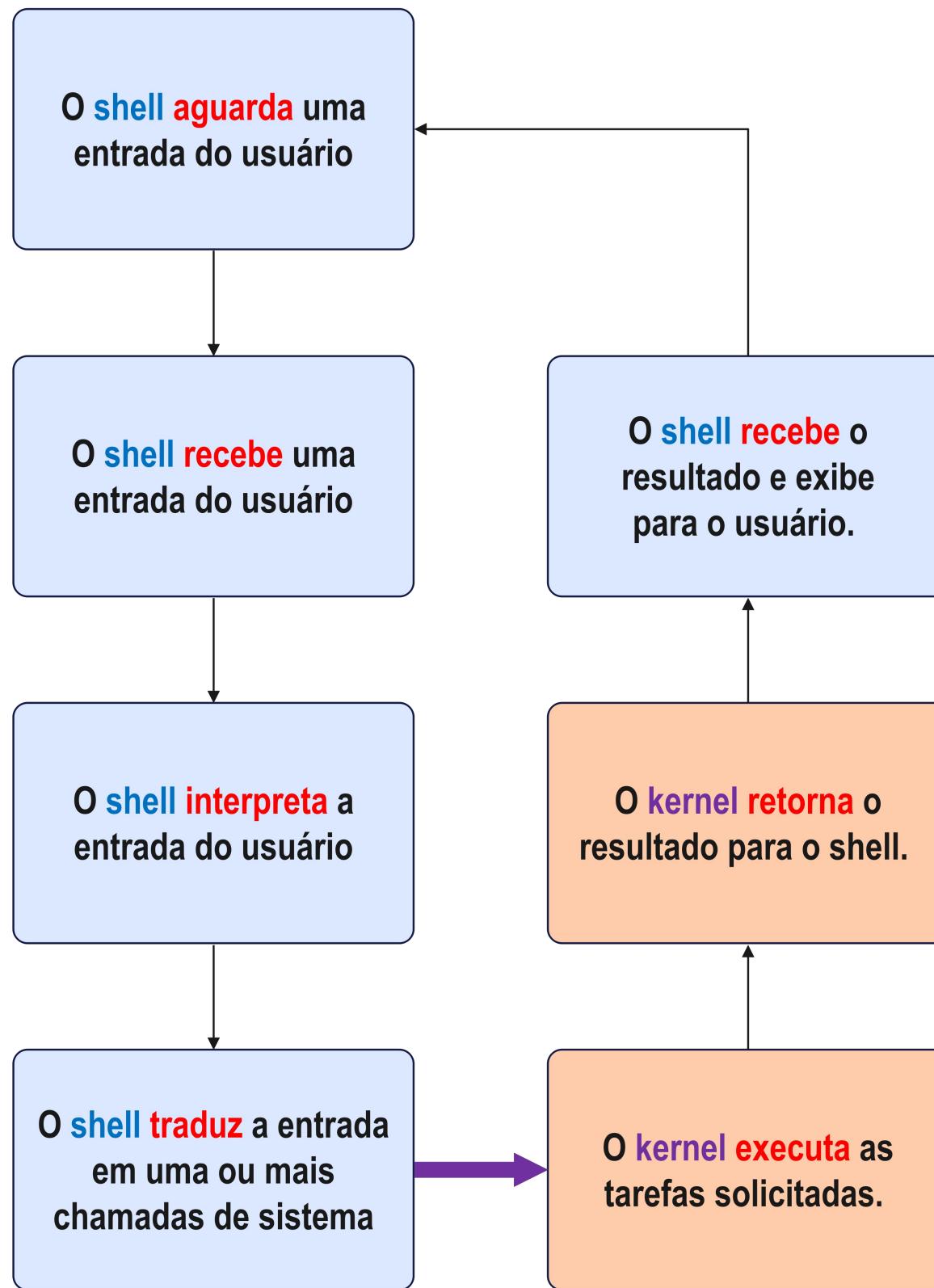
# O SHELL realiza chamadas de sistema (system calls)



Chamadas de sistema (system calls) são a linguagem que o kernel entende, um **conjunto bem definido de funções que os programs em modo usuário pode invocar para solicitar serviços do kernel**. Esse conjunto de funções é seguro.

Um programa de usuário NÃO pode acessar o disco para ler um arquivo, mas ele PODE "solicitar" ao shell para fazer uma chamada de sistema ao kernel, solicitando que o kernel leia o arquivo para ele.

# Como funcionam as chamadas de sistema?

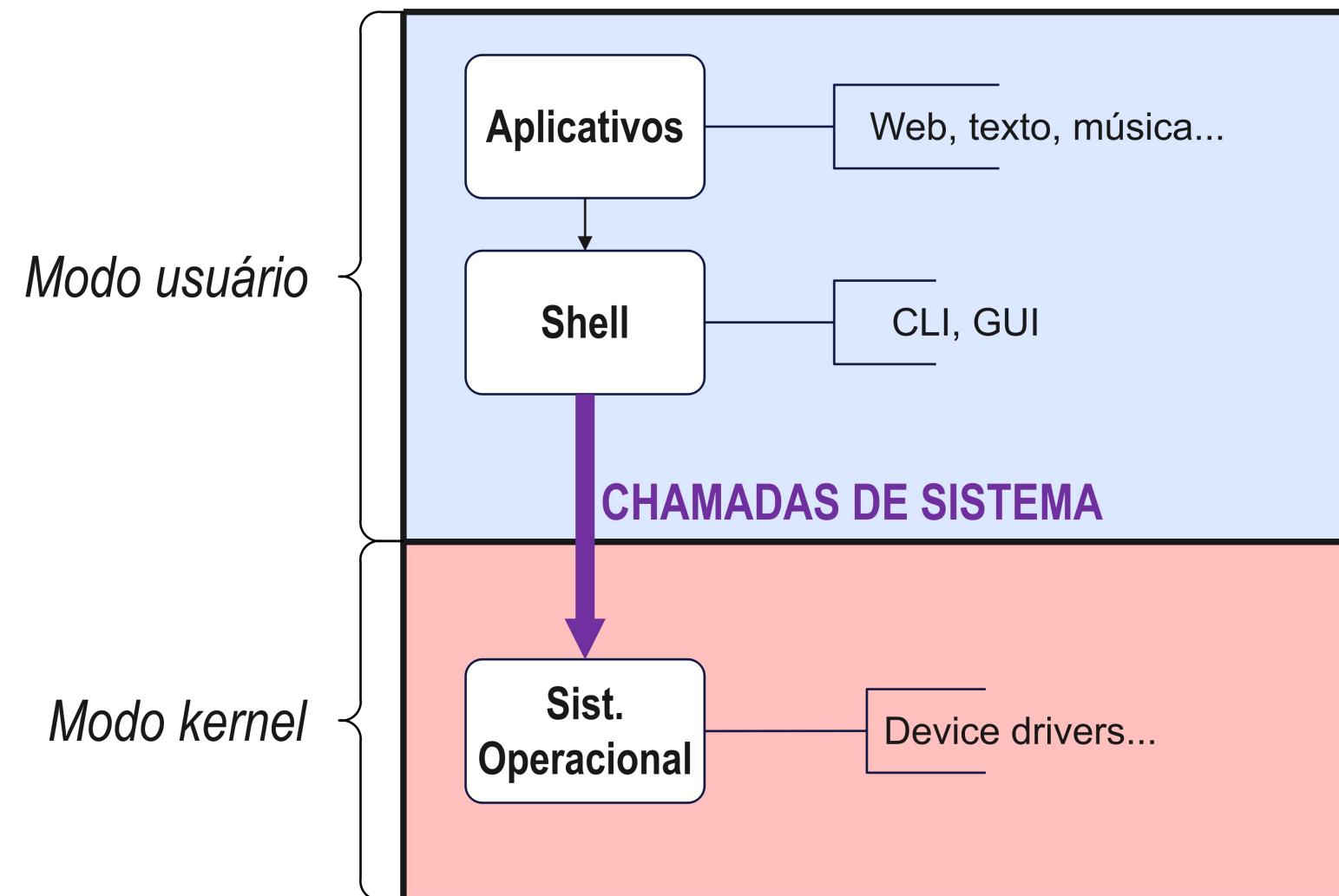


A entrada pode ser composta por diversas coisas diferentes, por exemplo:

- um comando digitado, por exemplo "`ls -l`"
- uma ação gráfica, por exemplo clicar no arquivo

O shell INTERPRETA a entrada do usuário, gera as chamadas de sistema adequadas e passa o controle para o kernel. O kernel executa as tarefas solicitadas e retorna o resultado para o shell. O shell exibe o resultado para o usuário.

# O que é o shell?

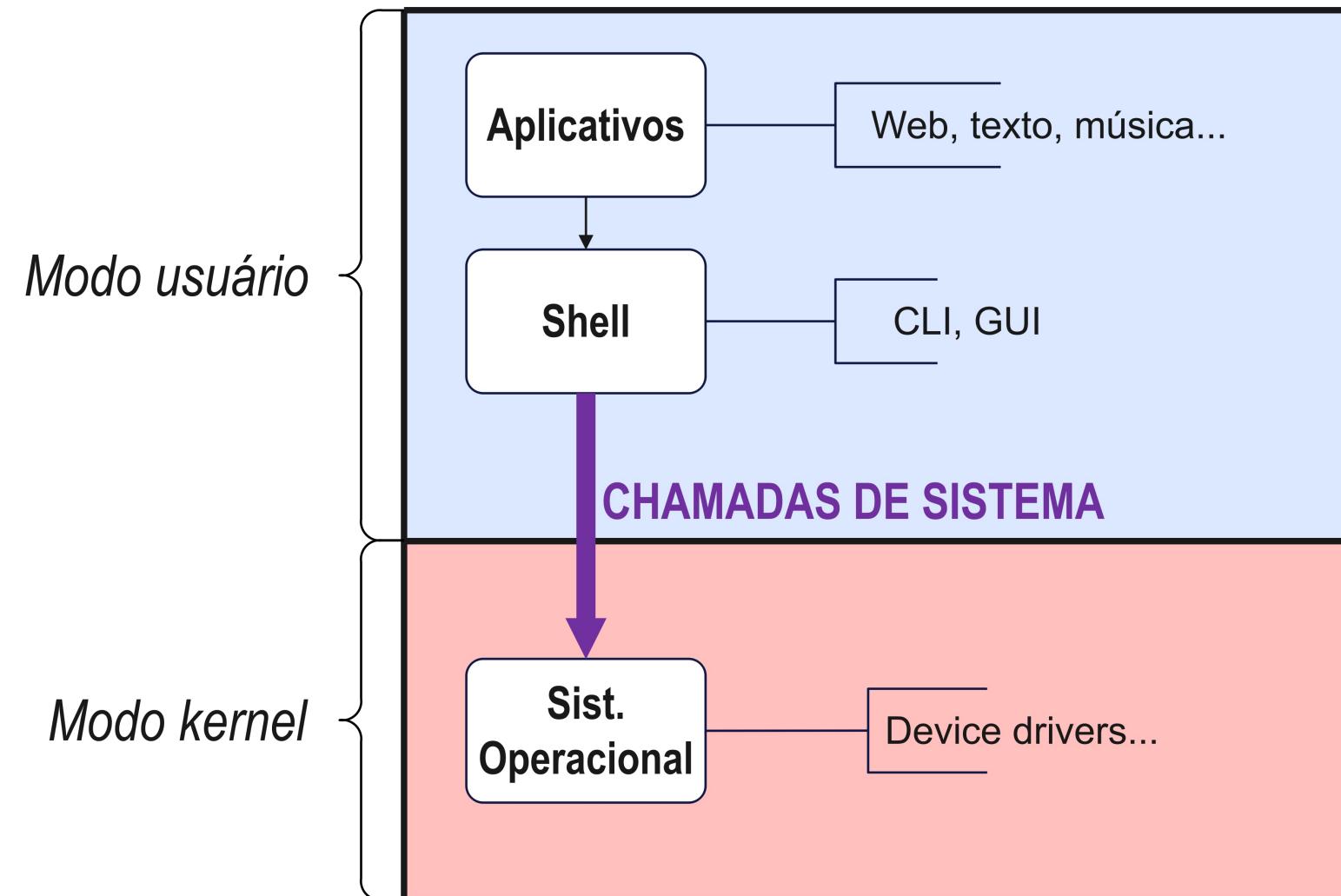


Ao tentar definir o que é o shell, a primeira coisa importante a notar é que **o shell não faz parte do sistema operacional!**

O shell é um programa de usuário como outro qualquer, com uma pequena diferença: ele recebe e interpreta as solicitações feitas pelos usuários ou outros programas, e gera as chamadas de sistema para o kernel.

A separação entre o modo kernel e o modo usuário, essencial para a estabilidade e segurança, cria a necessidade de um mecanismo de comunicação controlado (as chamadas de sistema). O shell é esse programa especializado cujo propósito é tornar essa comunicação acessível e utilizável para o usuário final.

# O que é o shell?

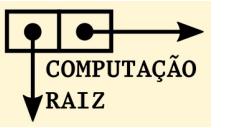


Uma coisa que causa confusão é entender que **shell é um papel funcional**, o de ser **uma interface para o sistema operacional**, em vez de ser uma única tecnologia ou programa específico.

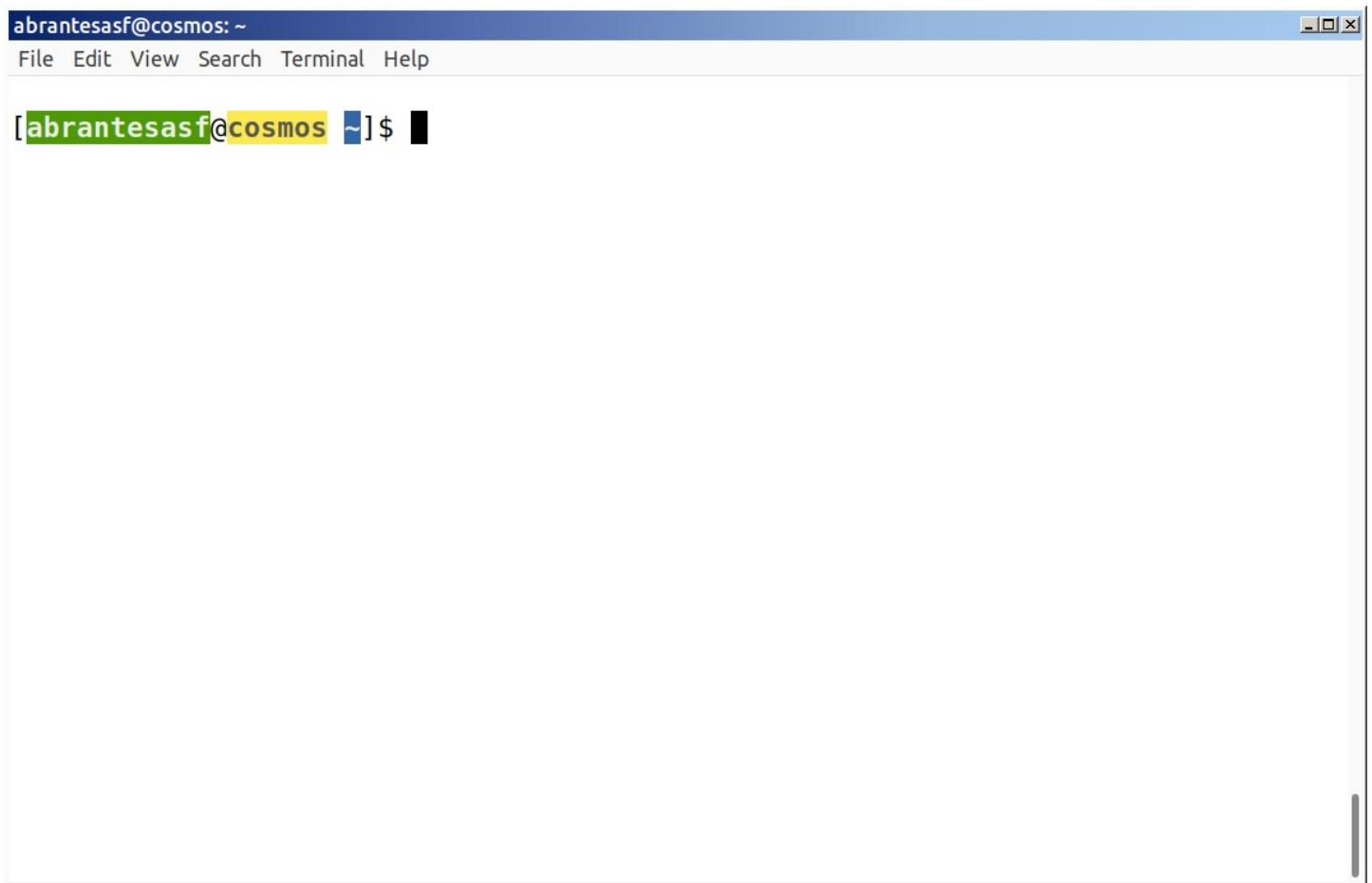
A forma como esse papel é implementado varia drasticamente entre diferentes sistemas e contextos. Em geral temos 3 "tipos" de manifestações da interface:

<b>CLI</b>	command-line interface
<b>GUI</b>	graphical user interface
<b>TUI</b>	text user interface

# Shell CLI: command-line interface

A screenshot of a terminal window titled "abrantesasf@cosmos: ~". The window has a blue header bar with standard window controls (minimize, maximize, close) on the right. Below the header is a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The main area of the terminal shows a command prompt: "[abrantesasf@cosmos ~]\$". A vertical scroll bar is visible on the right side of the terminal window.

# Shell CLI: command-line interface



abrantesASF@cosmos: ~

File Edit View Search Terminal Help

[abrantesASF@cosmos ~]\$ █

Um shell CLI tem uma interface baseada em texto na qual o usuário interage com o sistema operacional digitando comandos alfanuméricos.

O shell interpreta esses comandos e faz as chamadas de sistema necessárias, exibindo o resultado também em modo texto.

## Vantagens:

Acesso total ao poder do sistema operacional;  
Automação e scripting;  
Controle granular de todo o sistema;  
Eficiência de recursos (memória e processamento);  
Acesso remoto (SSH).

# Shell CLI: command-line interface



Alguns dos principais shells CLI para UNIX/Linux/Mac:

## Nome

- Thompson Shell (1971)
- Bourne Shell (1977)
- C Shell (1978)
- TC Shell (1983)
- **Korn Shell** (1983)
- **Bash** (1989)
- Almquist Shell (1989)
- **Z Shell** (1990)
- POSIX Shell (1992)
- Scheme Shell (1994)
- Fish (2005)
- Ion (2015)
- ...

## Comando:

- sh
- sh
- csh
- tcsh, sh
- ksh
- bash, sh
- sh
- zsh
- sh
- scsh
- fish
- ion

# Shell CLI: command-line interface



Alguns dos principais shells CLI para Windows:

**Nome**

- Command Prompt
- PowerShell

**Comando:**

**cmd.exe**

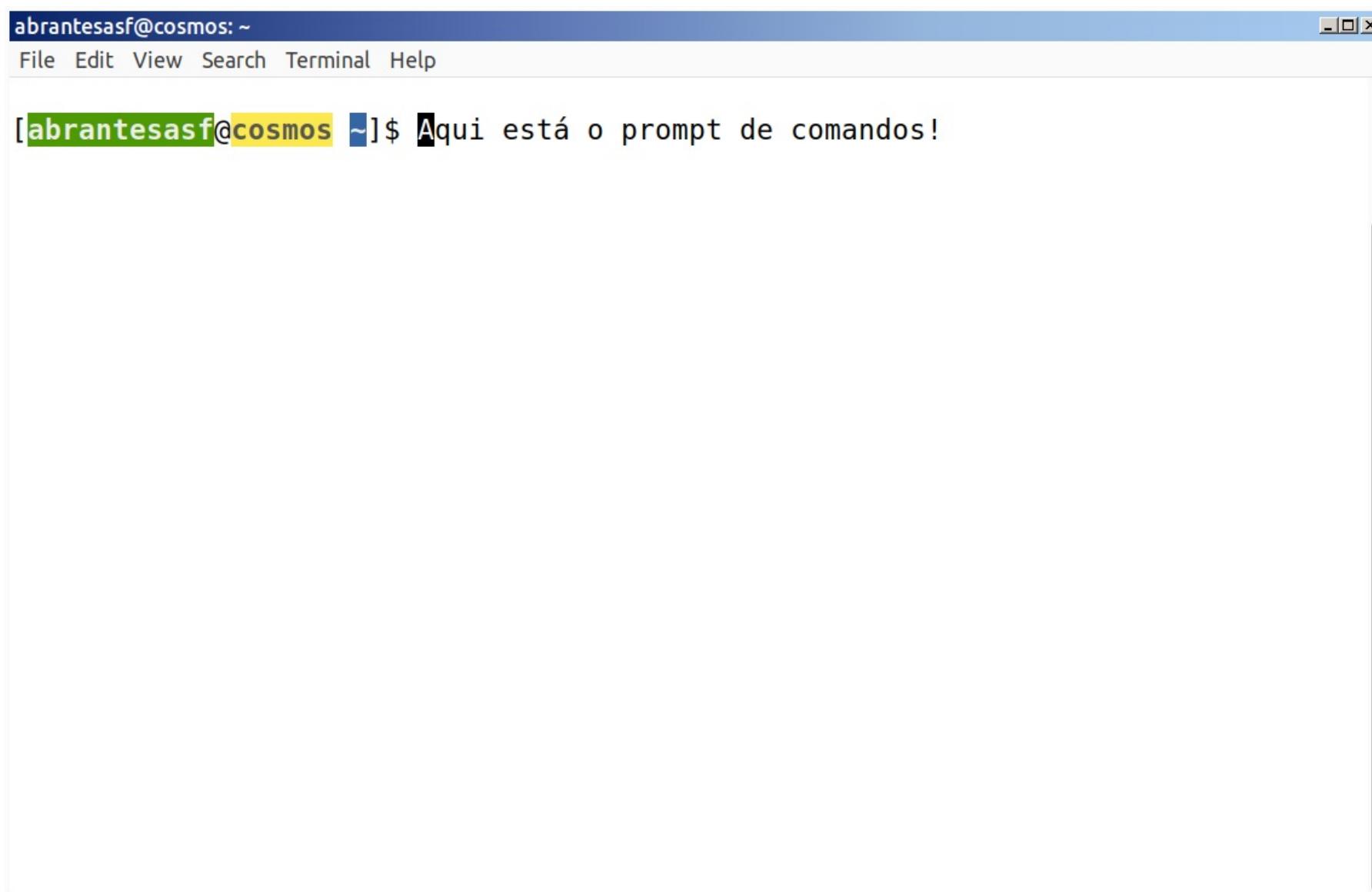
**powershell.exe**

```
PowerShell 7.5.0
PS C:\Users\John> $PSVersionTable

Name                           Value
----                          -----
PSVersion                      7.5.0
PSEdition                     Core
GitCommitId                    7.5.0
OS                             Microsoft Windows 10.0.22631
Platform                       Win32NT
PSCompatibleVersions           {1.0, 2.0, 3.0, 4.0...}
PSRemotingProtocolVersion      2.3
SerializationVersion           1.1.0.1
WSManStackVersion              3.0

PS C:\Users\John> function Say-Hello {
>> Write-Host "Hello, World!"
>>
PS C:\Users\John> Say-Hello
Hello, World!
PS C:\Users\John> |
```

# Shell GUI: graphical user interface



abrantesasf@cosmos: ~

File Edit View Search Terminal Help

[abrantesasf@cosmos ~]\$ Aqui está o prompt de comandos!

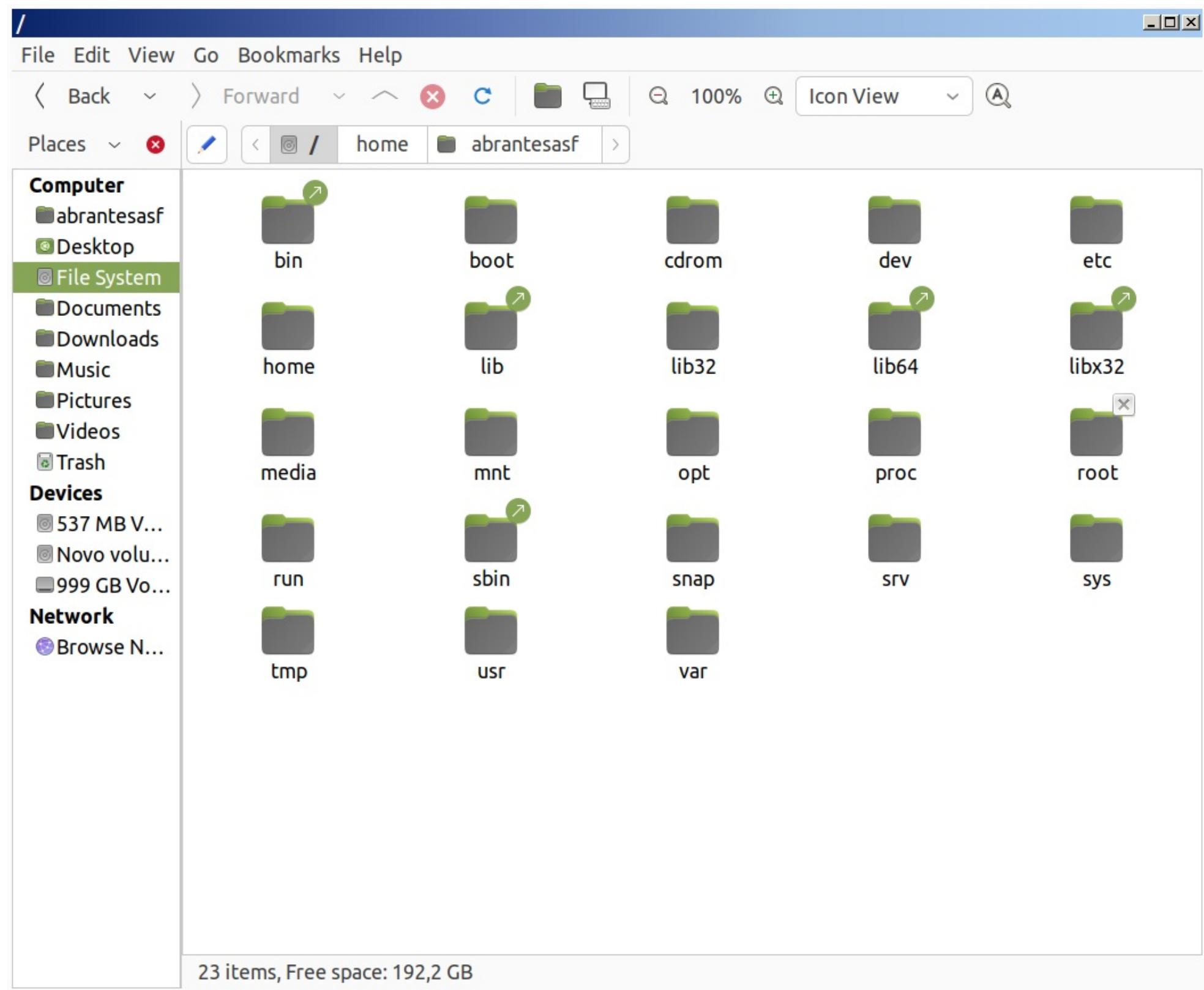
Um shell CLI tem um **prompt de comandos**, uma linha onde o usuário digita os comandos de interação com o sistema operacional.

Alguns comandos são do próprio shell, outros são programas separados.

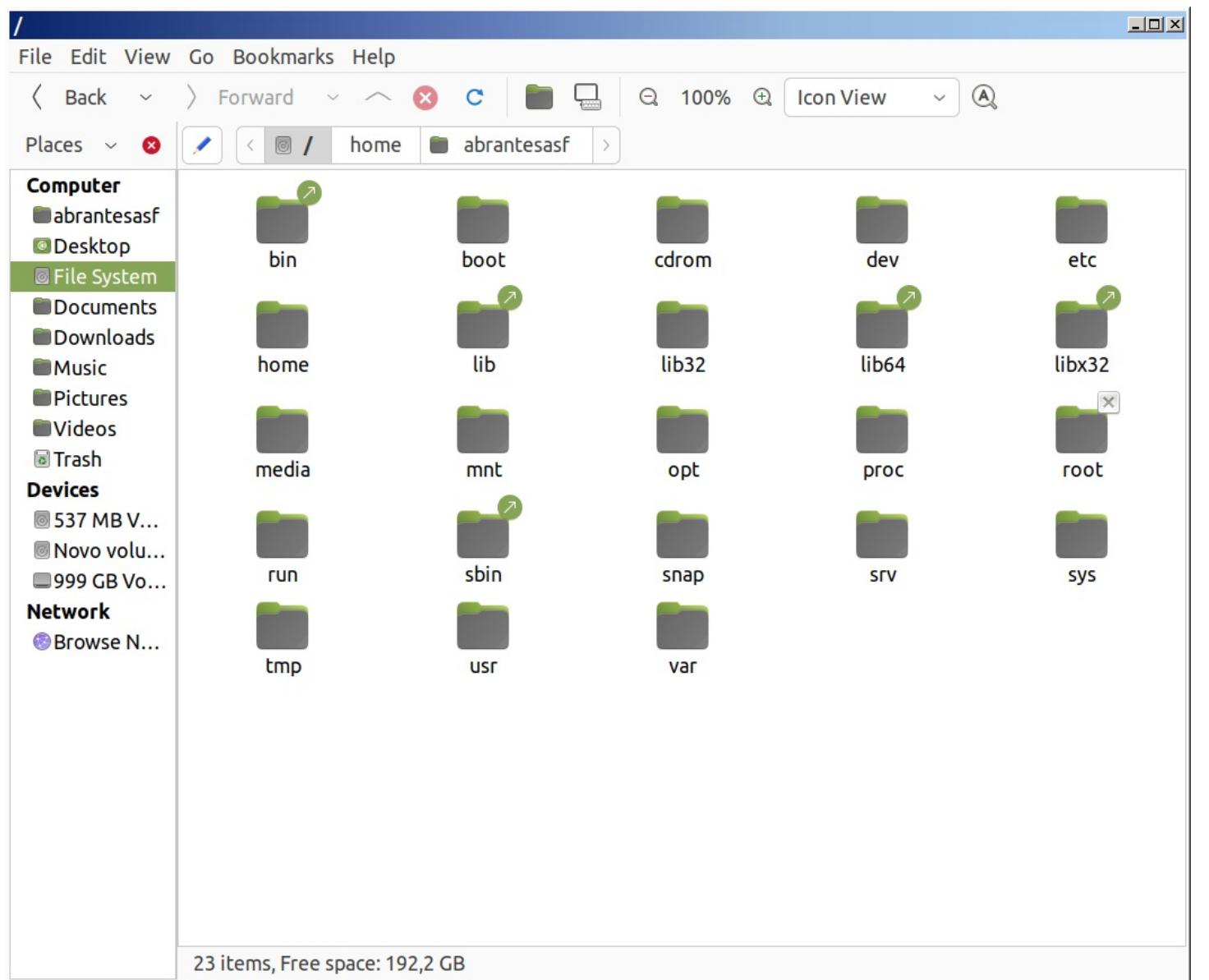
Permite criar scripts para automatizar tarefas.

O prompt é totalmente configurável. Em geral, se mostra "\$" é o prompt de usuário comum; se mostra "#" é o prompt do usuário root (o administrador do sistema).

# Shell GUI: graphical user interface



# Shell GUI: graphical user interface



Um shell GUI tem uma interface que utiliza elementos visuais (ícones, janelas, menus, botões, ponteiro, etc.) para permitir a interação com o sistema operacional. As ações são realizadas através de cliques do mouse, abertura de menus, arrastar e soltar, etc.

## Vantagens:

- Fácil aprendizagem e uso
- Feedback visual imediato
- Multitarefa visual

# Shell GUI: graphical user interface



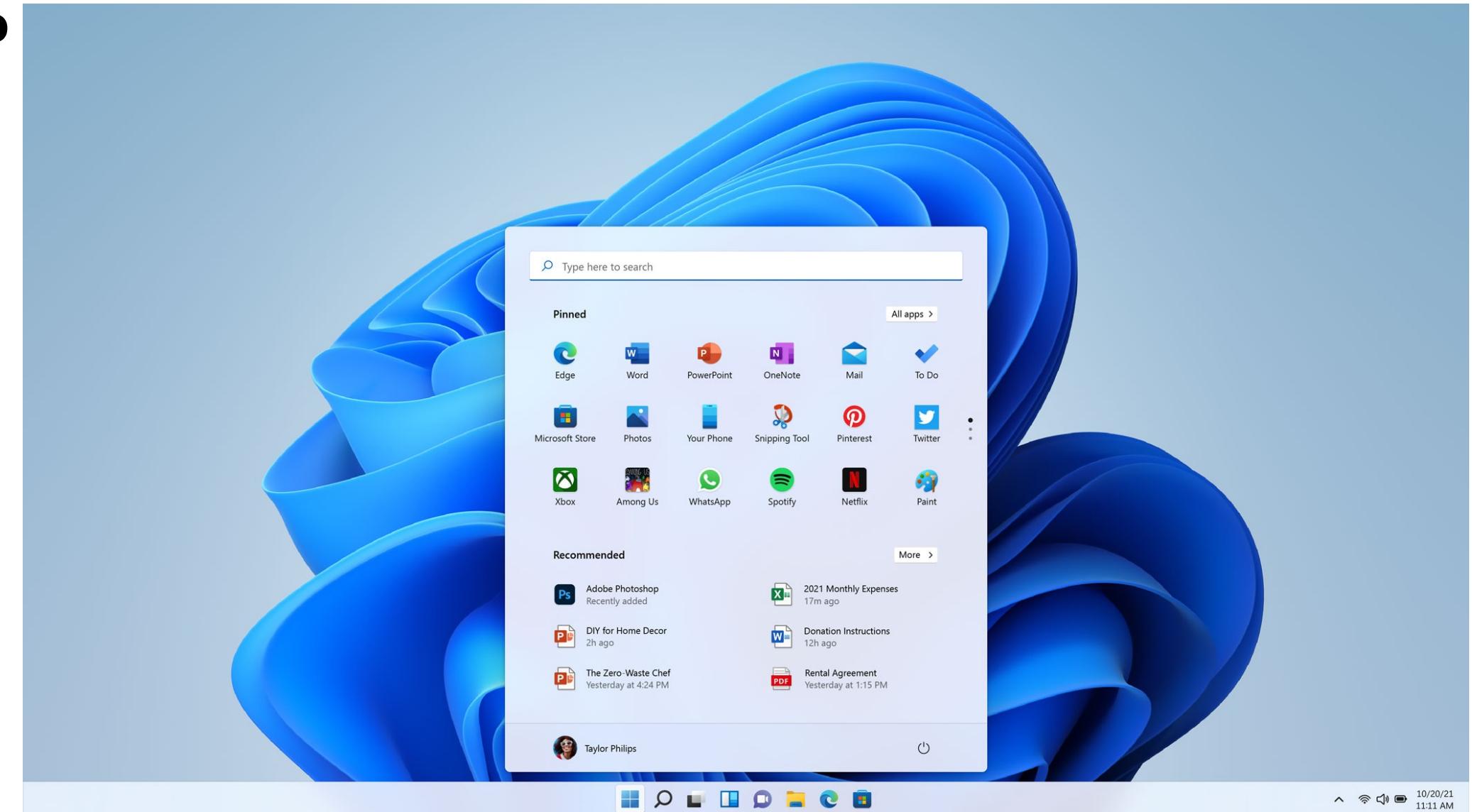
Alguns dos principais shells GUI: Windows

**Windows Shell (explorer.exe e outros)**

**Fornece a Área de Trabalho**

**Fornece o Menu Iniciar**

**Fornece a Barra de Tarefas**



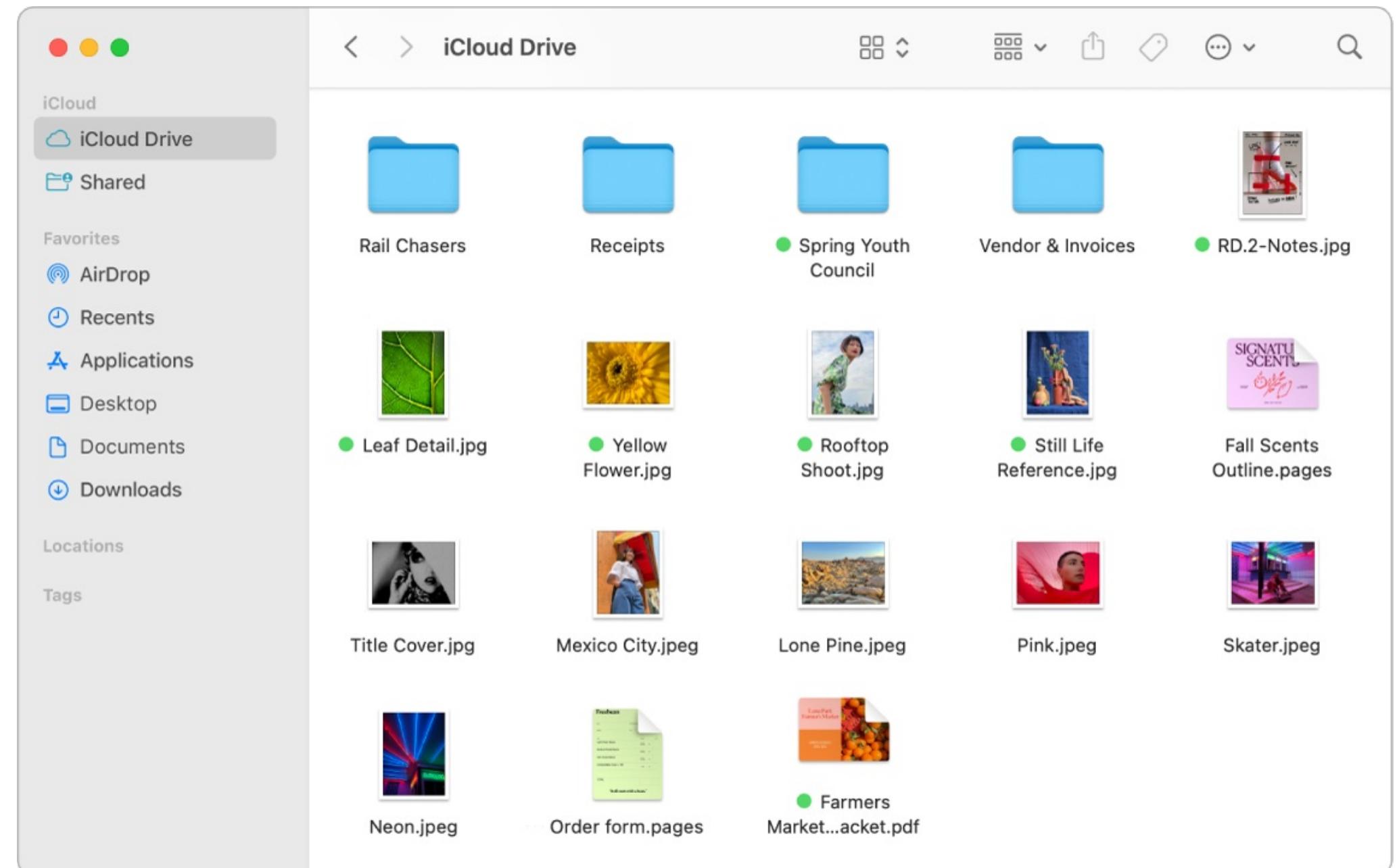
# Shell GUI: graphical user interface



Alguns dos principais shells GUI: Mac:

## Finder

Gestão de arquivos  
Experiência visual

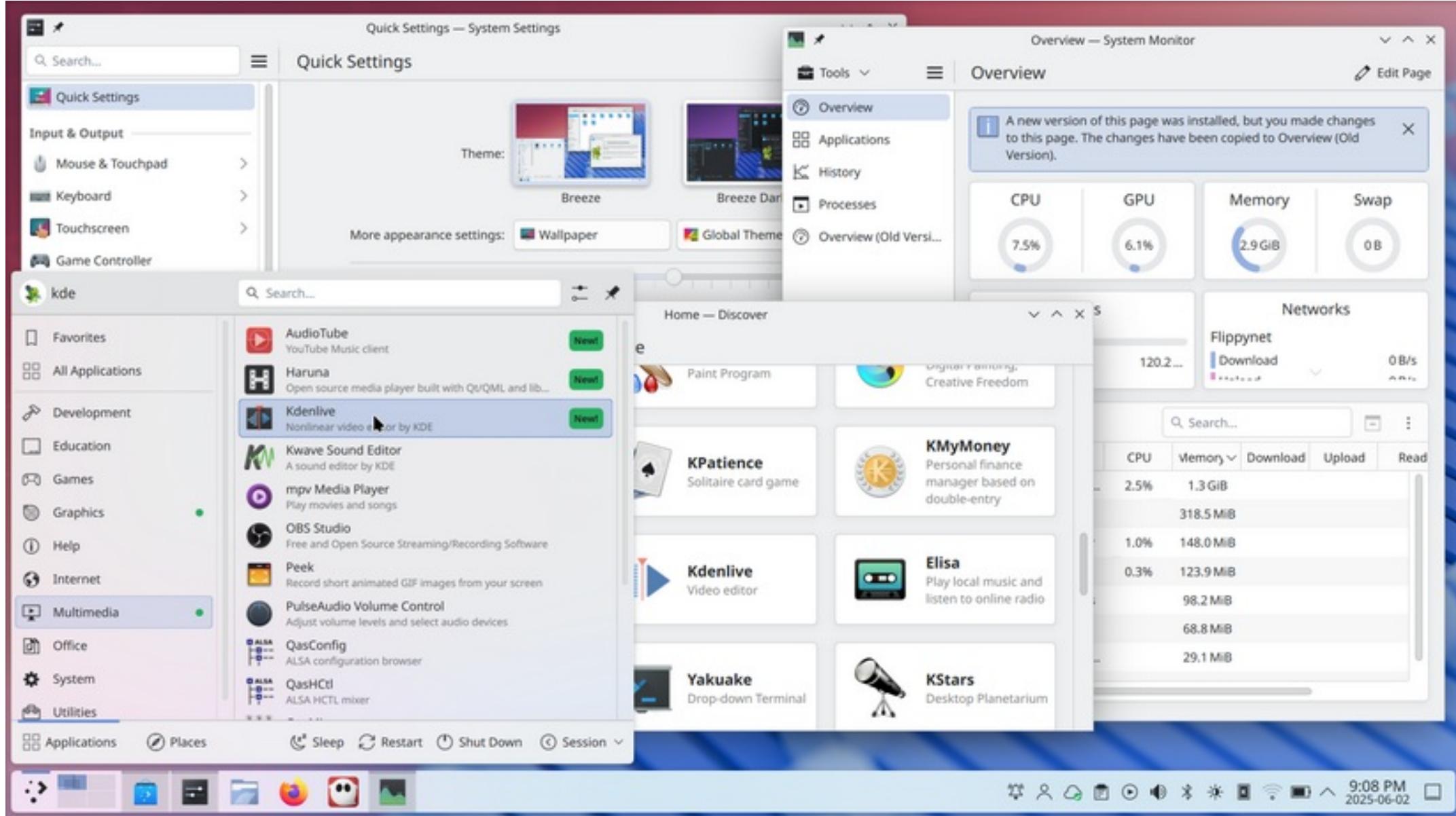


# Shell GUI: graphical user interface



## Alguns dos principais shells GUI: Linux

### KDE Plasma

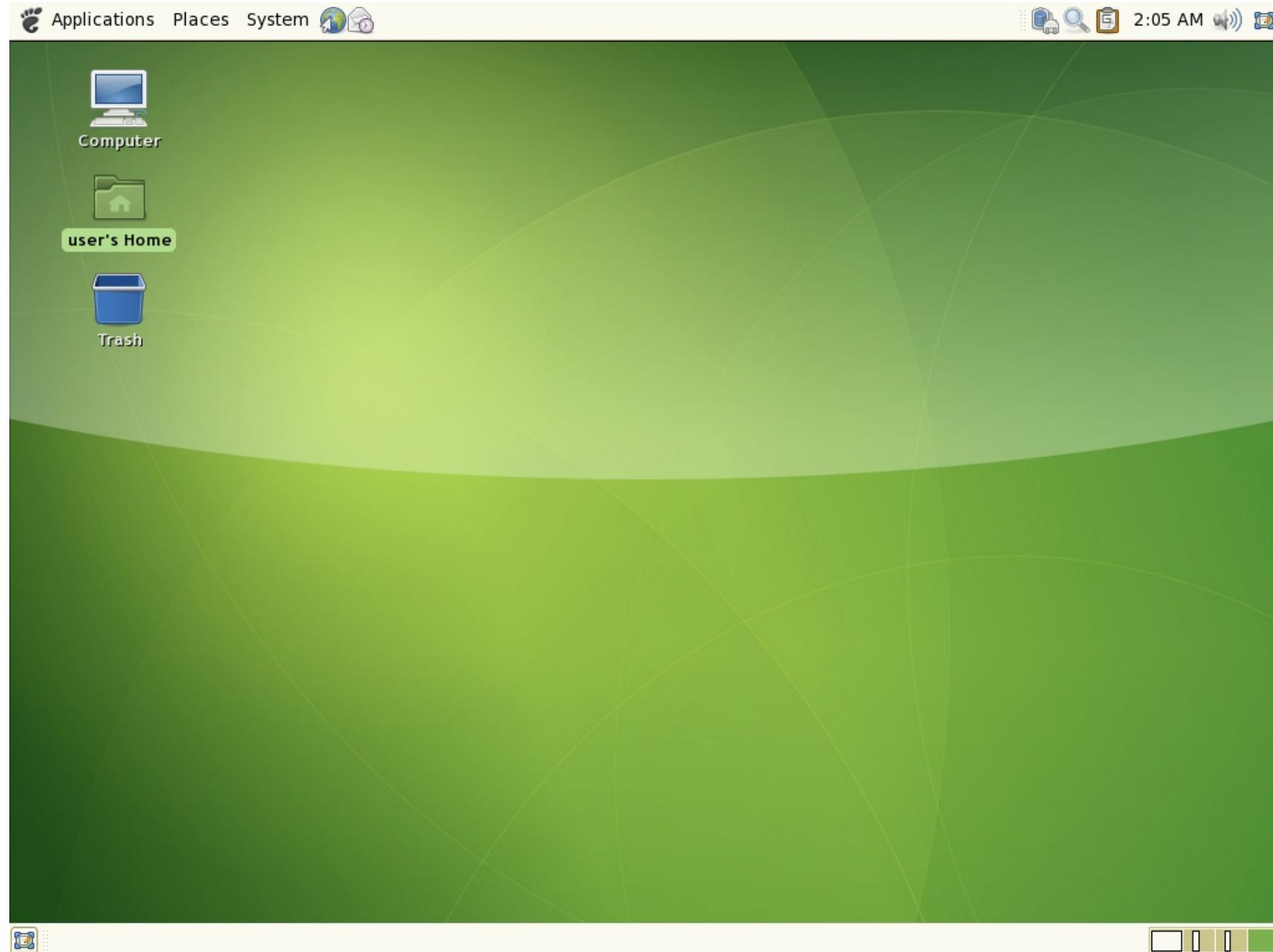


# Shell GUI: graphical user interface



Alguns dos principais shells GUI: Linux

Gnome 2

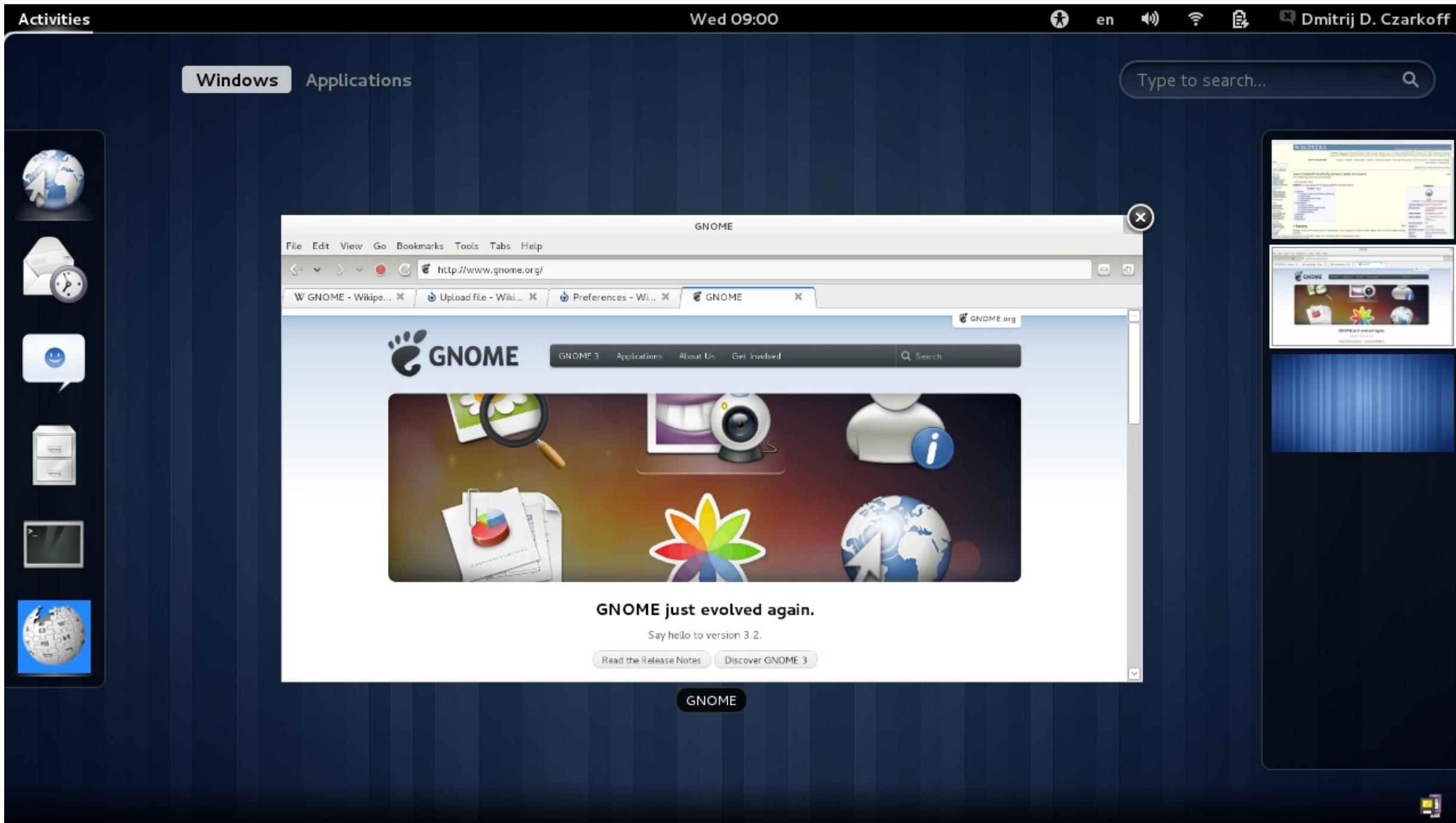


# Shell GUI: graphical user interface



## Alguns dos principais shells GUI: Linux

### Gnome 3

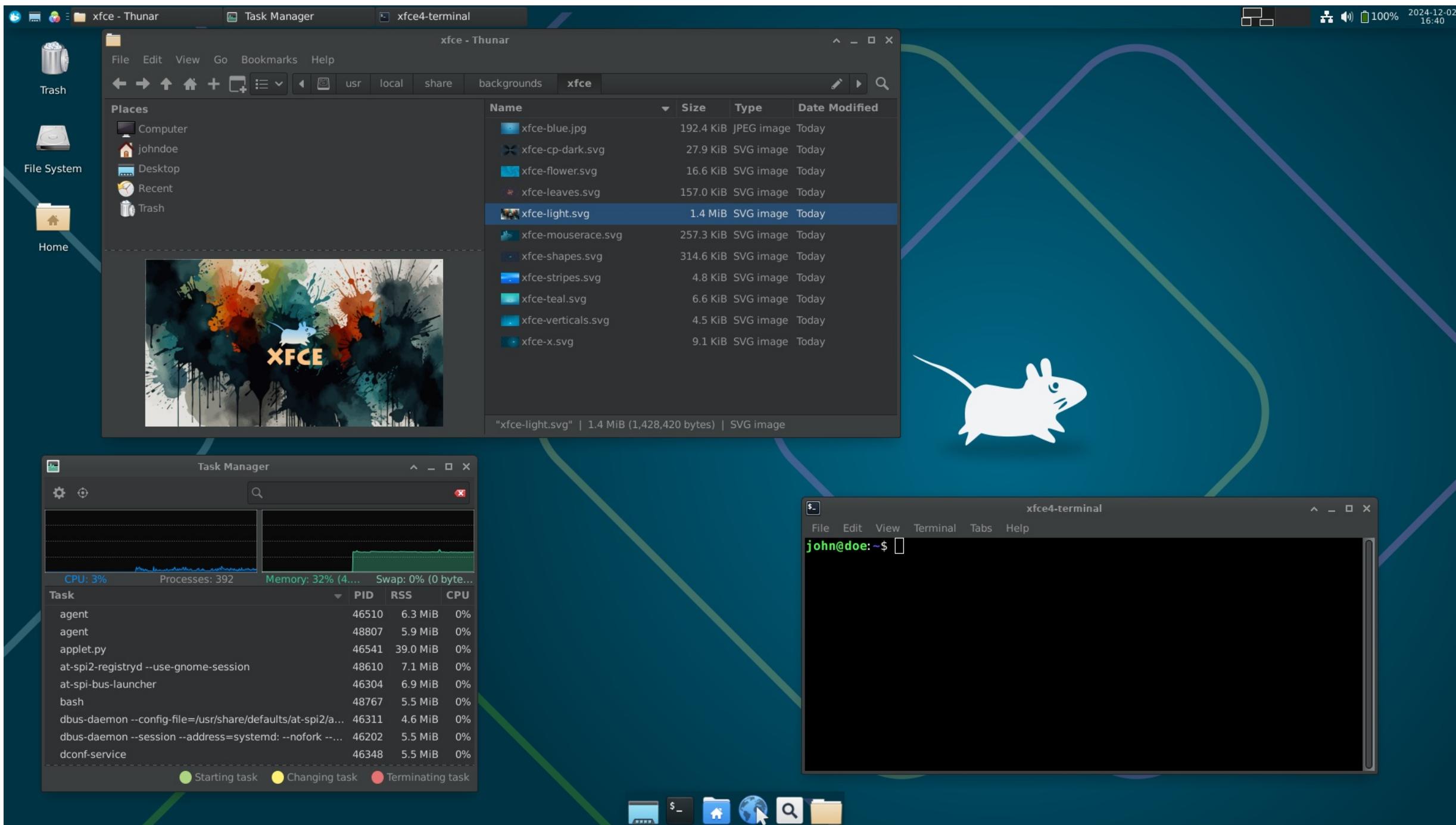


# Shell GUI: graphical user interface



## Alguns dos principais shells GUI: Linux

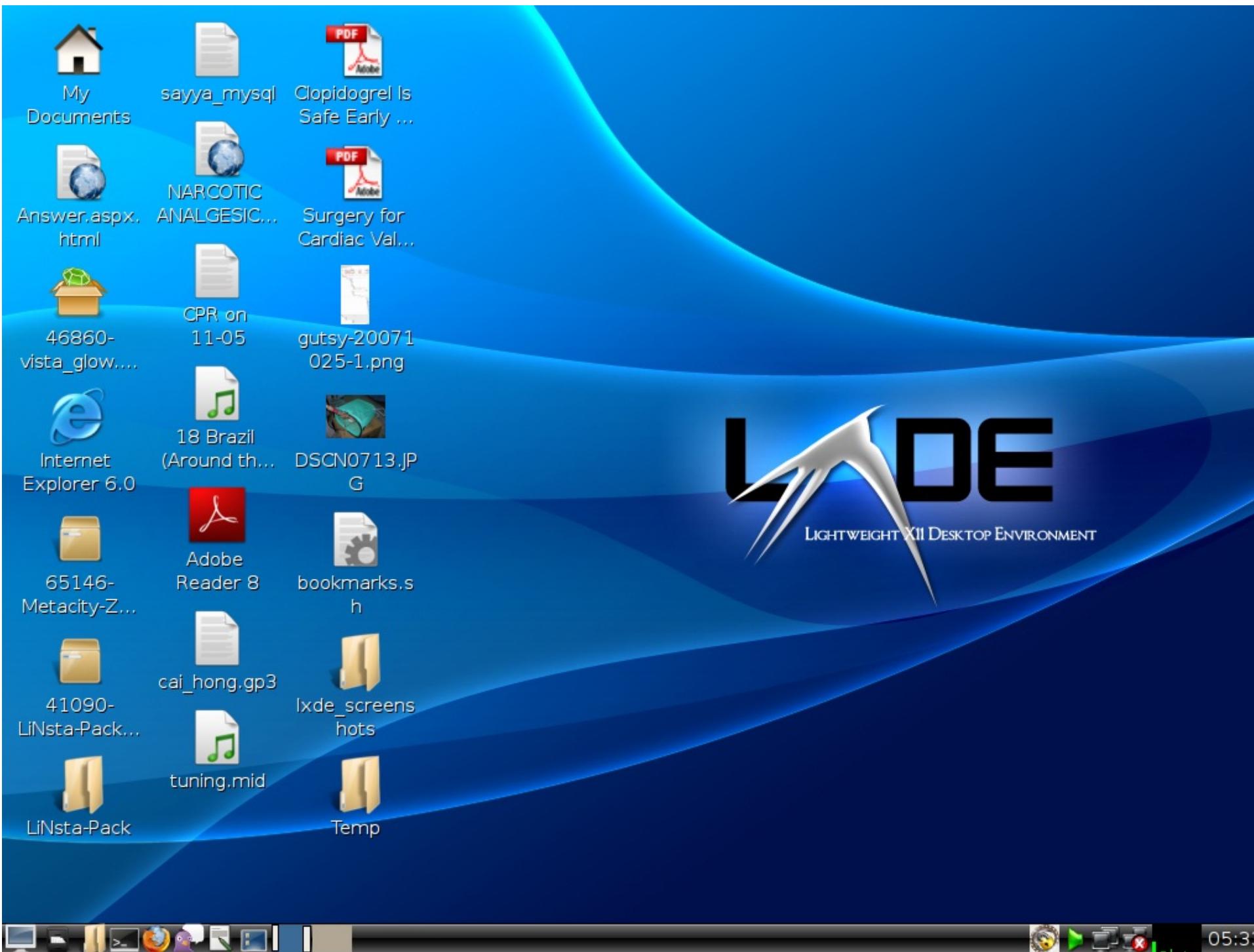
XFCE



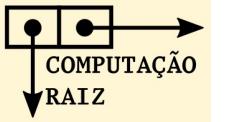
# Shell GUI: graphical user interface

## Alguns dos principais shells GUI: Linux

LXDE



# Shell GUI: graphical user interface



## Alguns dos principais shells GUI: Linux

MATE

The image shows two windows from the MATE desktop environment. On the left, the 'About MATE Desktop Environment' window is open, displaying information about the MATE desktop environment version 1.24.0. It includes a large green circular logo with a play button icon, the text 'MATE Desktop Environment', the version number '1.24.0', and a paragraph about the history of MATE. It also contains links to the 'Website' and 'Credits'. On the right, the 'Desktop User Guide' window is open, specifically showing the 'Global Shortcut Keys' section. This window lists various keyboard shortcuts with their functions, such as 'Alt + F1' for opening the Applications Menu and 'Ctrl + Alt + Arrow keys' for switching workspaces.

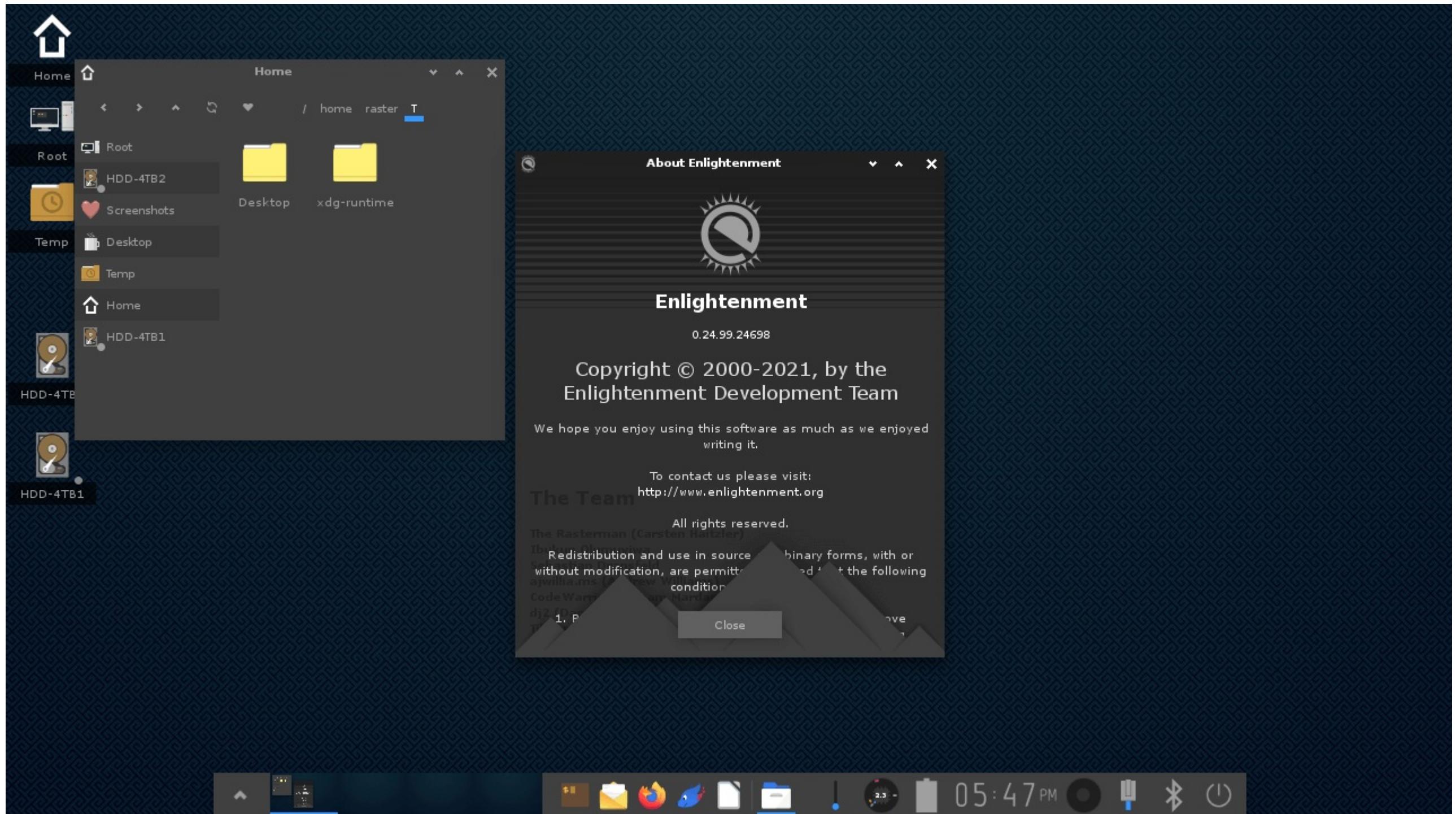
Shortcut Key	Function
Alt + F1	Open the Applications Menu.
Alt + F2	Display the Run Application dialog. See <a href="#">Running Applications</a> for more information.
Print Screen	Take a screenshot of the entire desktop. See <a href="#">Taking Screenshots</a> for more information.
Alt + Print Screen	Take a screenshot of the currently focused window.
Ctrl + Alt + Arrow keys	Switch to the workspace to the specified direction of the current workspace. See <a href="#">Workspaces</a> for more information on working with multiple workspaces.
Ctrl + Alt + D	Minimize all windows and give focus to the desktop.
Alt + Tab	Switch between windows. A list of windows that you can select is displayed. Release the keys to select a window. You can press the Shift key to cycle through the windows in reverse order.
Ctrl + Alt + Tab	Switch the focus between the panels and the desktop. A list of items that you can select is displayed. Release the keys to select an item. You can press the Shift key to cycle

# Shell GUI: graphical user interface



## Alguns dos principais shells GUI: Linux

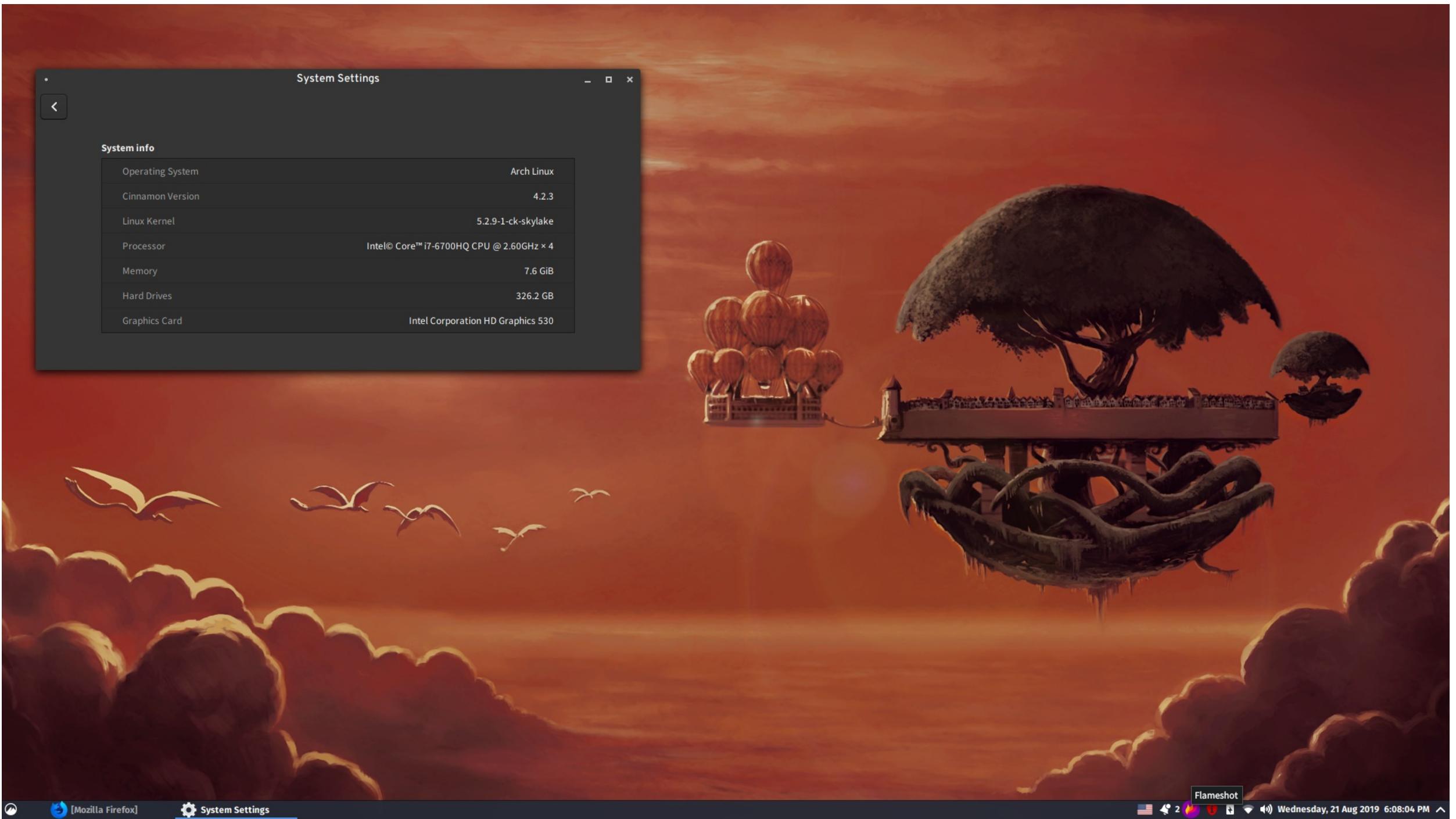
### Enlightenment



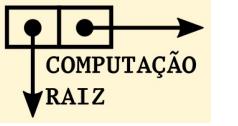
# Shell GUI: graphical user interface

## Alguns dos principais shells GUI: Linux

### Cinnamon



# Shell GUI: graphical user interface



Alguns dos principais shells GUI: Linux

Pantheon:



# Shell GUI: graphical user interface



## Alguns dos principais shells GUI: Linux

### i3 Window Manager

The screenshot shows a desktop environment using the i3 Window Manager. In the top-left corner, there's a terminal window titled "handlers.c" in Vim, displaying C code for handling window map requests. Below it is another terminal window showing a transcript of a video player (MPlayer) playing a file from "/Series/The.Big.Bang.Theory/S01". A video player window is also visible, showing a scene from the TV show "The Big Bang Theory". At the bottom of the screen, there's a dock with several icons, including a file browser, a terminal, and links to the i3 website and GitHub.

**i3 - an improved dynamic tiling window manager**

Goals Documentation Downloads Contact Impressum/Imprint

i3 was created because wml, our favorite window manager at the time, didn't provide some features we wanted (Xinerama done right, for example), had some bugs, didn't progress since quite some time and wasn't easy to hack at all (source code comments/documentation completely lacking). Still, we think the wml developers and contributors did a great job. Thank you for inspiring us to create i3.

Please be aware that i3 is primarily targeted at advanced users and developers.

### Goals

Based upon the experiences we made when wanting to hack/fix wml, we agreed upon the following goals for i3:

1. Write well readable, well **documented** code. Create additional documentation on how to extend i3 by explaining its internal workings.  
This includes being modifiable by people who do know how to program but who are not necessarily familiar with all of X11's internals. That is, document why things happen and when they happen so that the user gets a picture of the whole process a Window Manager is responsible of by just reading the source code.
2. Use xcb as far as possible (it does not provide functions for some features yet, like XKB) instead of Xlib. xcb has a much cleaner API and should be faster in quite a lot of situations.
3. Implement Xinerama correctly, that is by assigning each workspace to a virtual screen. Especially make sure that attaching and detaching new monitors like video projectors works during operation and does the right thing.
4. Use the metaphor of a table for abstraction. You can create horizontal and vertical columns (in wml, you can only create vertical columns). See the documentation for

<http://i3.zekjur.net/> [7/7] Top Tor Disabled

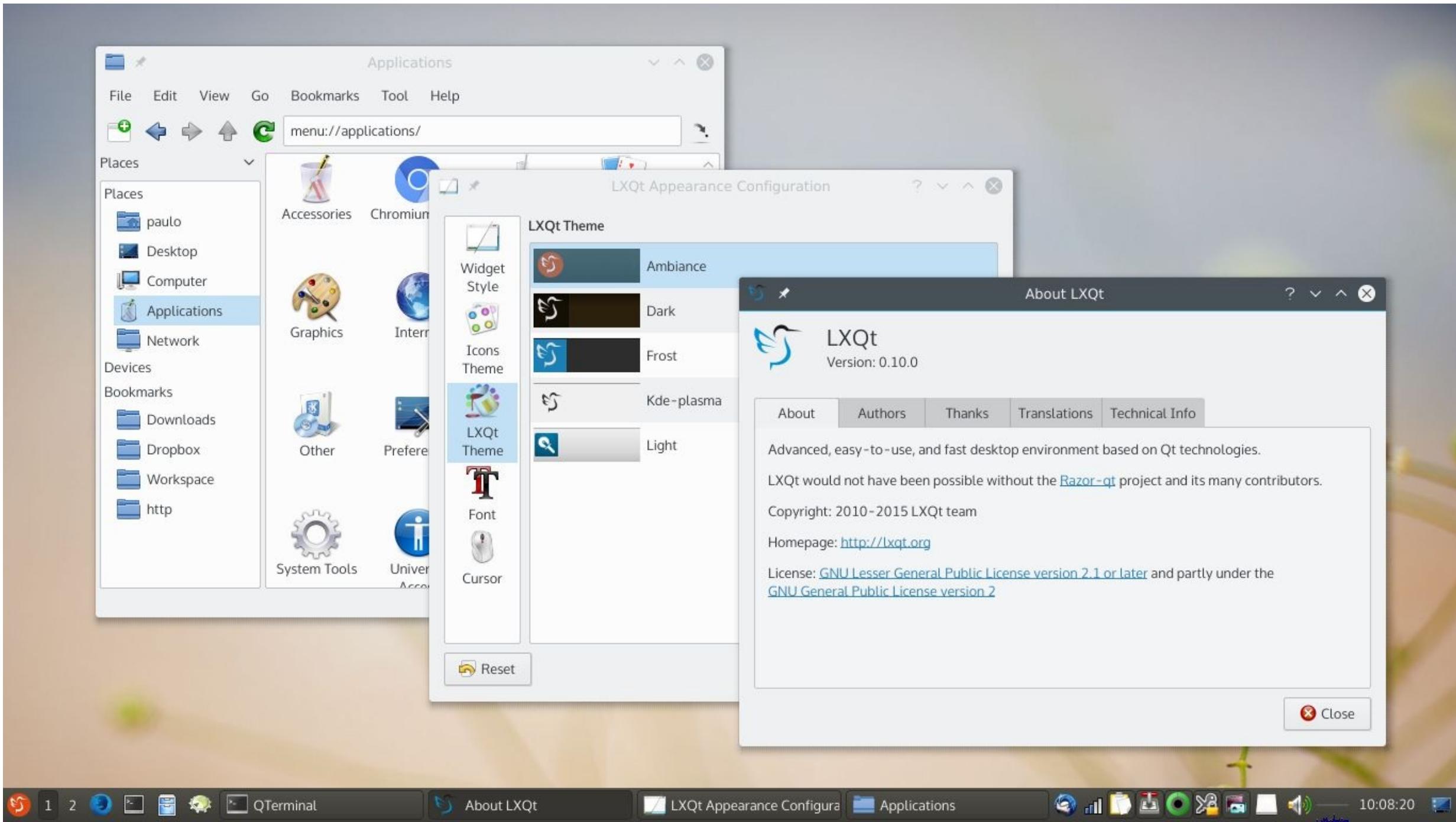
DHCP: no | VPN: no | W: down | E: 192.168.1.42 | FULL 90.74% 00:00:00 | 0.09 0.08 0.09 | 16.03.2009 09:52:16

# Shell GUI: graphical user interface



## Alguns dos principais shells GUI: Linux

LXQt

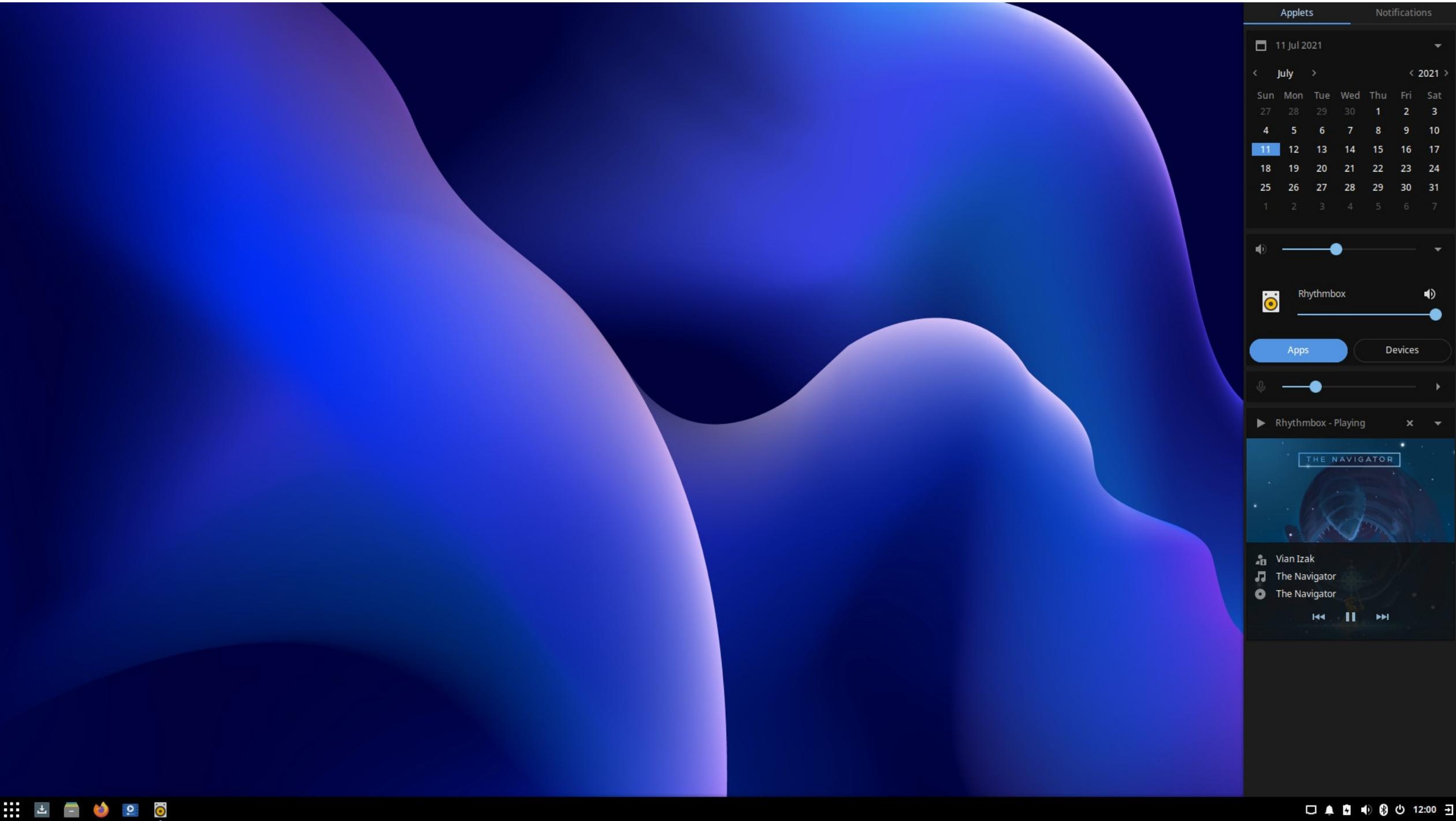


# Shell GUI: graphical user interface



Alguns dos principais shells GUI: Linux

Budgie:



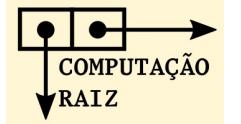
# Shell GUI: graphical user interface

Alguns dos principais shells GUI: Linux

Existem inúmeros outros shells GUI para Linux, consulte a Internet se quiser conhecer as dezenas de outras opções disponíveis.

Obs.: no Linux há uma distinção entre ambiente gráfico, gerenciadores de janela e Shells GUI mas, aqui, para simplificar, estou considerando tudo como shell GUI. No futuro veremos a diferença.

# Shell TUI: text user interface



Left	File	Command	Options	Right
/software			/etc	
	Name	Size	MTime	Name
	/..	4096	Oct 2 04:02	/..
	/ICAClient-3.0	2048	Jan 6 2003	/.java
	/aida-2.1.1	2048	Apr 28 2003	/ada
	/amber-6.0	2048	Feb 27 2004	/conf
	/amber-7.0	2048	Mar 5 2004	/config
	/amber-7.0p	2048	Apr 16 2004	/cron.d
	/amber-8	2048	Dec 22 2004	/default
	~ansys61	34	Jan 7 2003	/dt
	~ansys71	34	Nov 28 2003	/fscklogs
	/ant-1.6	2048	Aug 10 13:26	~fstyp.d
	/apache-1.3.27	2048	Dec 16 2002	~httpd
	/apache-1.3.28	2048	Jan 6 2004	/init.d
	/apache-1.3.33	2048	Feb 7 2005	/js
	/autoconf-2.57	2048	May 27 2004	/lost+found
	/autodock-305	2048	Jan 5 2001	/mail
	/ICAClient-3.0			/cron.d
Hint: Keys not working in xterms? Use our xterm.ad, .ti and .tcap files.				
aisa:/software> \$ [^]				
1Help 2Menu 3View 4Edit 5Copy 6RenMov 7Mkdir 8Delete 9PullDn 10Quit				

# Shell TUI: text user interface

Left	File	Command	Options	Right		
/software				/etc		
Name	Size	MTime		Name	Size	MTime
..	4096	Oct 2 04:02		..	4096	Oct 2 04:02
/ICAClient-3.0	2048	Jan 6 2003		/.java	30	May 13 2004
/aida-2.1.1	2048	Apr 28 2003		/ada	4096	Aug 9 2001
/amber-6.0	2048	Feb 27 2004		/conf	151	Jul 19 2000
/amber-7.0	2048	Mar 5 2004		/config	4096	Dec 13 2004
/amber-7.0p	2048	Apr 16 2004		/cron.d	133	Sep 29 20:23
/amber-8	2048	Dec 22 2004		/default	75	Aug 12 2004
~ansys61	34	Jan 7 2003		/dt	27	Apr 5 2003
~ansys71	34	Nov 28 2003		/fscklogs	39	Aug 3 2000
/ant-1.6	2048	Aug 10 13:26		~fstyp.d	15	Apr 25 2000
/apache-1.3.27	2048	Dec 16 2002		~httpd	20	Jul 19 2000
/apache-1.3.28	2048	Jan 6 2004		/init.d	4096	Sep 21 15:45
/apache-1.3.33	2048	Feb 7 2005		/js	4096	Aug 9 2001
/autoconf-2.57	2048	May 27 2004		/lost+found	4096	Oct 8 2004
/autodock-305	2048	Jan 5 2001		/mail	4096	May 2 10:04
/ ICAClient-3.0				/cron.d		

Hint: Keys not working in xterms? Use our xterm.ad, .ti and .tcap files.

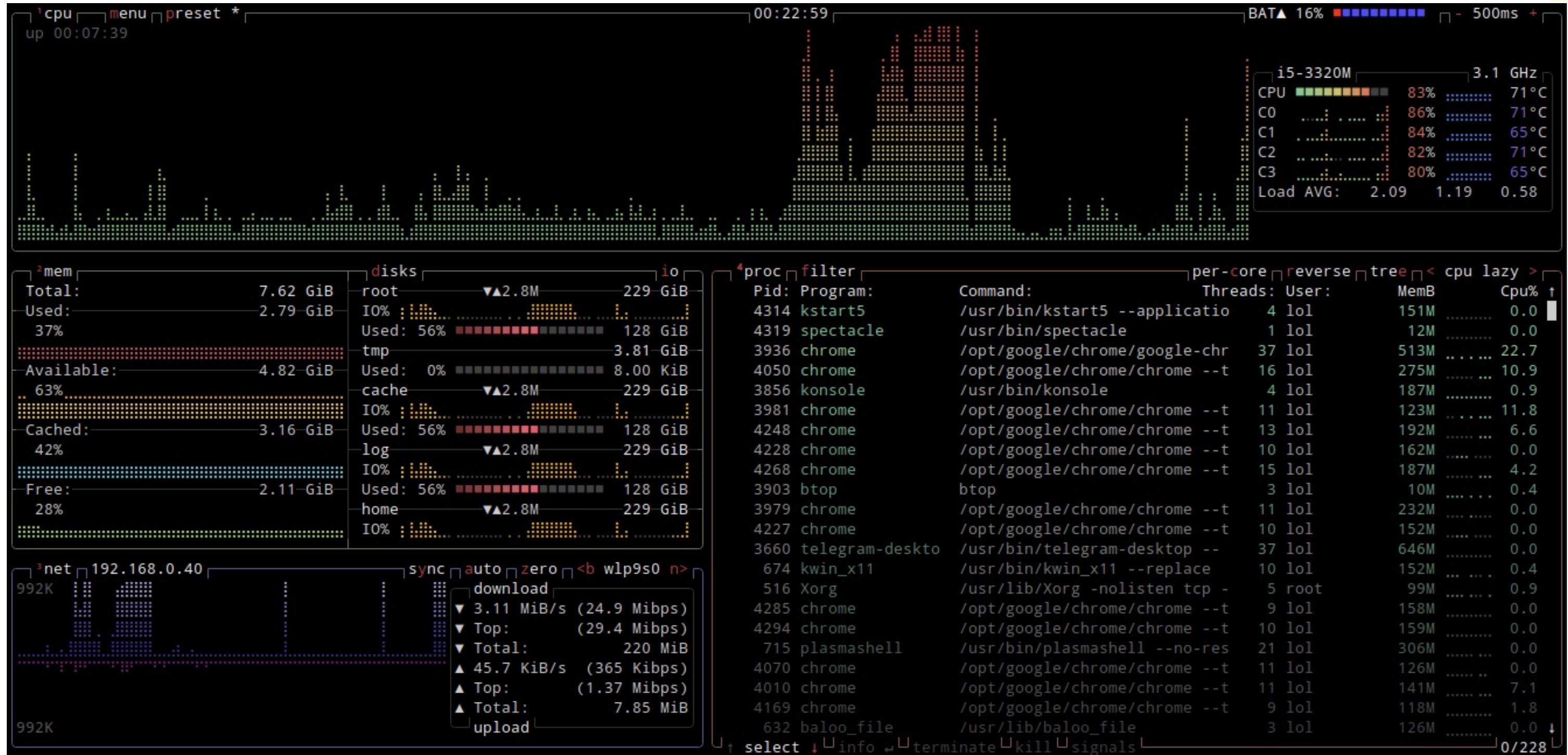
aisa:/software> \$ [^]

1Help 2Menu 3View 4Edit 5Copy 6RenMov 7Mkdir 8Delete 9PullDn 10Quit

**Um shell TUI é um intermediário entre um shell CLI e um shell GUI.**

**Os shells TUI utilizam caracteres em um terminal de texto CLI, mas criam uma interface que imita uma GUI, com janelas, botões e menus. Em alguns é possível usar o mouse normalmente.**

# Shell TUI: text user interface



Um shell TUI pode ser muito interessante para tarefas administrativas, como verificação do status do sistema, configuração do sistema e outras.

# Shell CLI x GUI

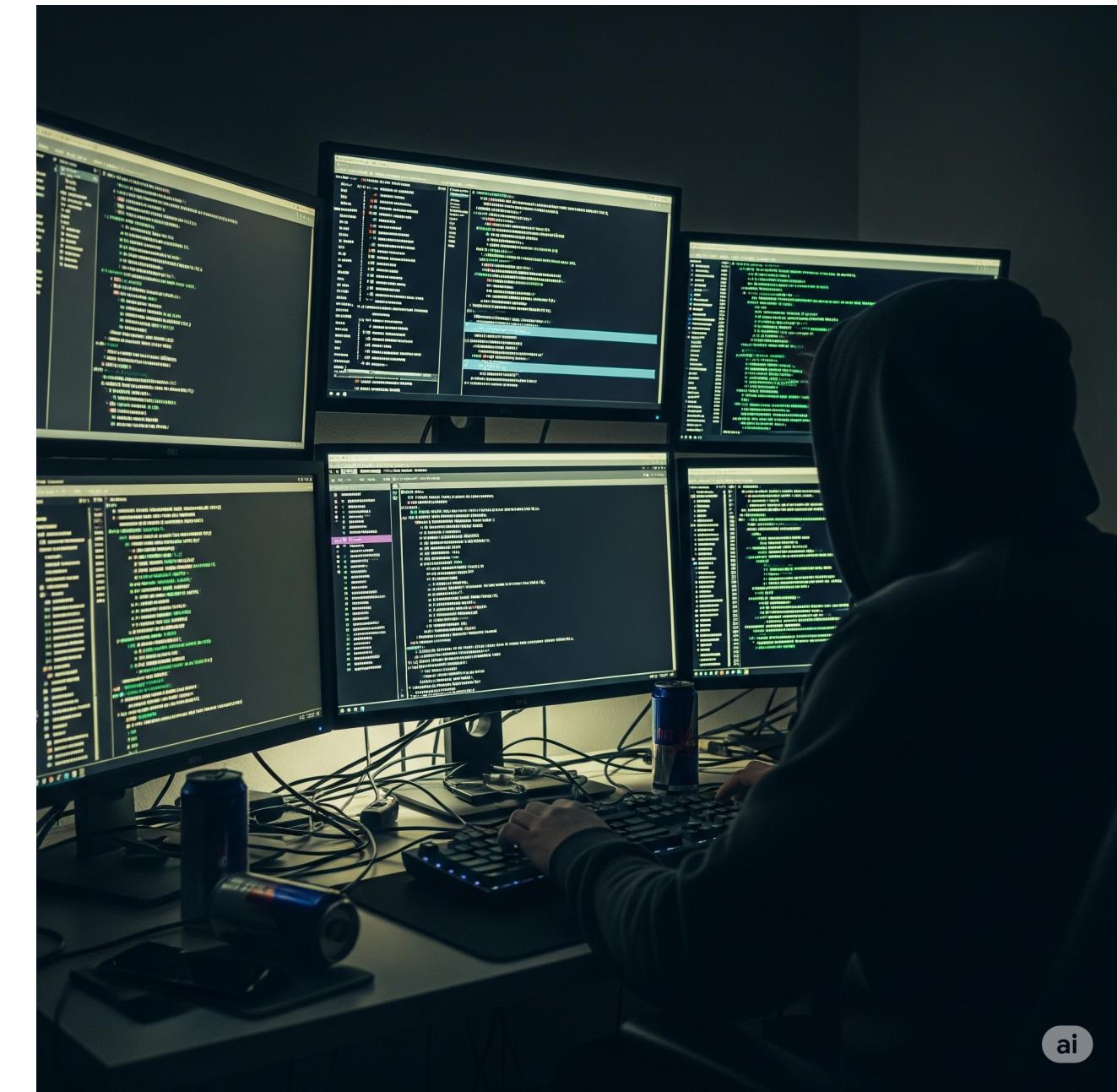


Característica	Shell CLI	Shell GUI
Interface	Baseada em texto	Visual, baseada em gráficos e ícones
Método de interação	Comandos digitados via teclado	Clique, arrastar/soltar, mouse
Curva de aprendizagem	Alta, necessita de memorização de comandos	Pequena, intuitiva para iniciantes
Consumo de recursos	Baixo (memória e CPU)	Alto (memória e CPU)
Poder de automação	Muito alto, através de scripts	Muito baixo, limitado a recursos específicos restritos
Flexibilidade e controle	Controle total e detalhado sobre o sistema	Muito limitado, restrito apenas às funcionalidades expostas
Utilidade	Administração de sistemas, programação, acesso remoto, automação	Uso geral como desktop, tarefas do dia a dia, usuário comum

# Shell CLI x GUI



ChatGPT



Gemini

# Confusão de termos: shell, console, terminal, prompt...



**Hoje em dia há uma grande confusão a respeito dos termos corretos para nomear coisas como o shell, console, terminal, terminal remoto e outros conceitos.**

**Esses termos não são sinônimos, apesar de hoje em dia serem utilizados de modo "intercambiável". A melhor maneira para entender a diferença entre esses termos é acompanhar a evolução histórica dos computadores.**

**Usaremos como modelo o IBM System/360.**

# Confusão de termos: shell, console, terminal, prompt...



Brochura de marketing do IBM System/360  
Foto: <https://www.ibm.com/history/system-360>

Na década de 1960 a **computação era totalmente centralizada**, com máquinas como o **mainframe IBM System/360**.

Nessa época os computadores eram recursos colossais, imensos, ocupando salas inteiras.

# Confusão de termos: shell, console, terminal, prompt...



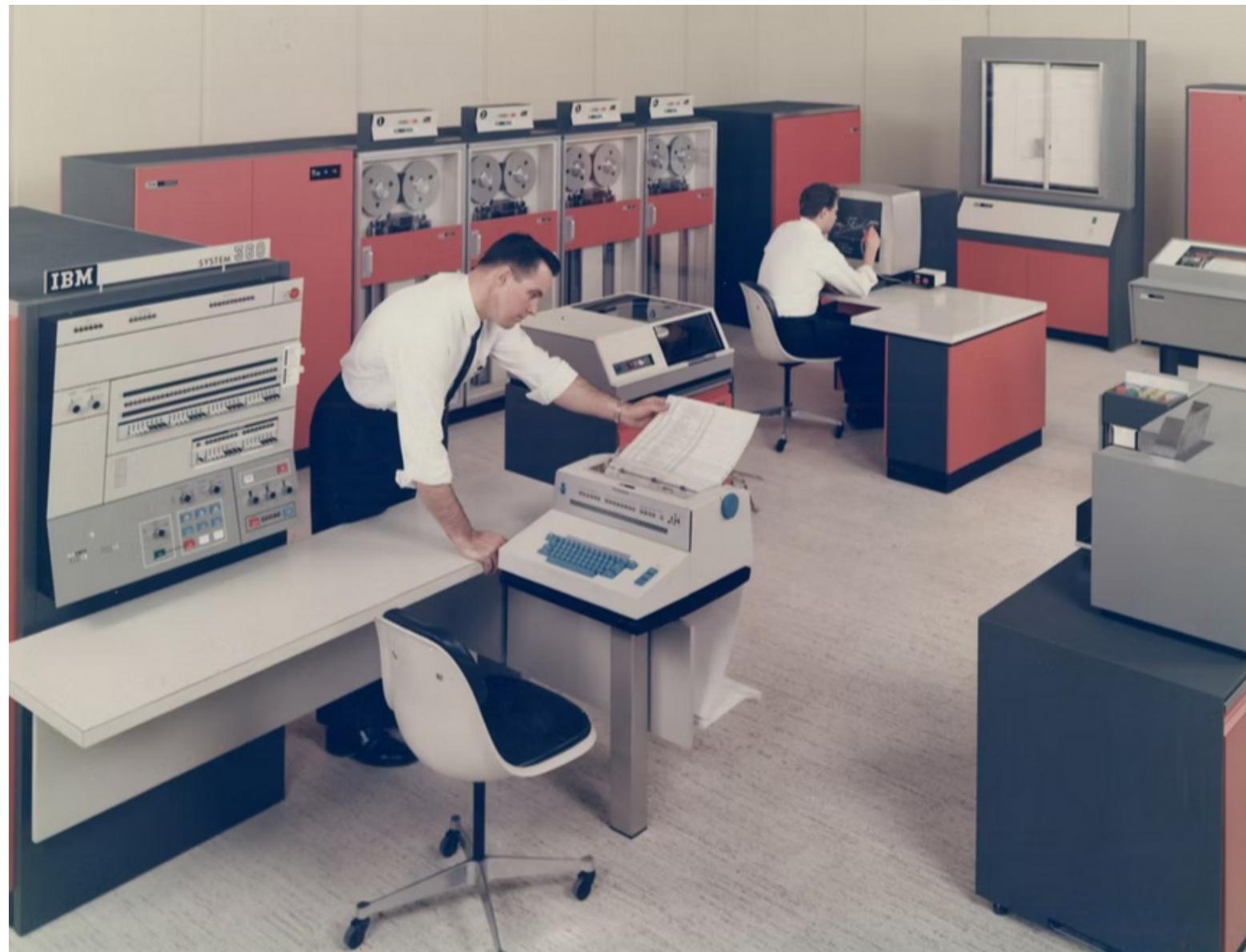
Sala com um IBM System/360  
Foto: <https://www.ibm.com/history/system-360>

Os computadores não eram pessoais, eram ativos corporativos compartilhados com dezenas ou centenas de pessoas.

A sala onde o computador ficava tinha acesso absolutamente controlado e apenas uns poucos usuários autorizados podiam entrar.

Esses usuários eram os **administradores do computador**.

# Confusão de termos: shell, console, terminal, prompt...



Sala com um IBM System/360

Foto: <https://www.ibm.com/history/system-360>

O papel dos **administradores do computador** era gerenciar, monitorar, diagnosticar e manter o estado físico e a "saúde" da própria máquina.

Eles não usavam as aplicações que o computador fornecia, eles garantiam que o computador estivesse funcional para que outros usuários pudessem usar.

Não havia monitor, teclado ou mouse. Como os administradores gerenciavam o computador então?

# Confusão de termos: shell, console, terminal, prompt...



Console do IBM System/360 Model 22

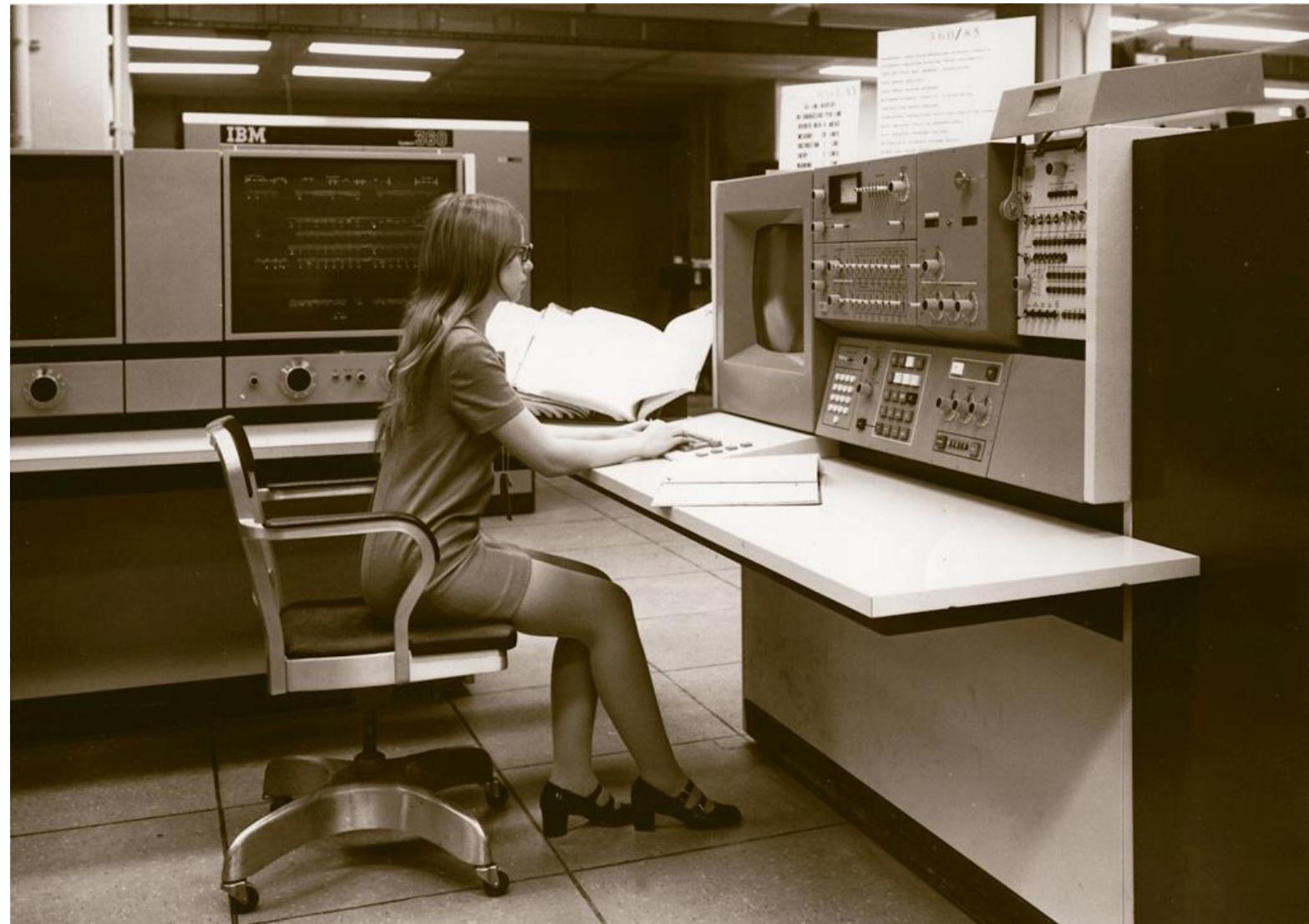
Foto: <https://www.righto.com/2019/04/iconic-consoles-of-ibm-system360.html>

Através do **CONSOLE** do sistema. O console era um painel de controle, uma interface direta com o hardware, com um poder praticamente ilimitado e absoluto sobre a máquina.

O console era parte integrante e fisicamente acoplada ao complexo da CPU e, através de painéis, luzes, alavancas, botões e seletores, fornecia um mapa físico do estado interno do computador.

Não era usado para digitação de programas ou textos, só para o gerenciamento do próprio computador.

# Confusão de termos: shell, console, terminal, prompt...



Console do IBM System/360 Model 85

Foto: [https://sellugsk.live/product\\_details/40558427.html](https://sellugsk.live/product_details/40558427.html)

**Os consoles foram evoluindo e alguns, como o IBM System/360 Model 85, tinha até um monitor para exibir mensagens do sistema e de diagnósticos.**

**Mas o console era usado apenas para a administração, por usuários administradores de sistema altamente especializados.**

**A entrada na sala do computador era rigidamente controlada pois não havia segurança nenhuma no console: qualquer pessoa com acesso ao console podia fazer qualquer coisa!**

# Confusão de termos: shell, console, terminal, prompt...



Console do IBM System/360 Model 91, NASA  
Foto: <https://en.wikipedia.org/wiki/File:360-91-panel.jpg>

# Confusão de termos: shell, console, terminal, prompt...



Aakash Dhage, no Unsplash  
<https://unsplash.com/photos/a-shiny-question-mark-on-a-purple-background-IFTGrwo6hN0>

**E os usuários? Como eles conseguiam acesso ao computador? Através dos TERMINAIS.**

**Um terminal era um dispositivo remoto que permitia que usuários (que NÃO estavam na sala do computador) tivessem acesso ao computador para rodar trabalhos e aplicações.**

**Um terminal era uma porta de entrada externa ao mainframe.**

# Confusão de termos: shell, console, terminal, prompt...



Teletype Model 33 ASR

Wikipedia ([https://en.wikipedia.org/wiki/File:ASR-33\\_1.jpg](https://en.wikipedia.org/wiki/File:ASR-33_1.jpg))

Os primeiros terminais utilizados eram dispositivos conhecidos como **TELETIPO**, ou TTY. Eram dispositivos eletromecânicos, essencialmente uma máquina de escrever robusta que podia ser controlada remotamente. Se conectava ao mainframe através de uma linha serial, muitas vezes usando um modem e linha telefônica comum.

A **experiência do usuário era baseada em papel**. O usuário digitava um comando, que era impresso na fita; ao teclar ENTER o comando era enviado ao mainframe; a resposta do mainframe era então transmitida de volta e impresso na linha seguinte.

Era lento e não tinha exibição dinâmica, mas mantinha uma cópia em papel de toda a sessão de trabalho. Modelos avançados tinham leitores e perfuradores de fita de papel, que servia como armazenamento de dados e entrada de programas "offline".

# Confusão de termos: shell, console, terminal, prompt...



DEC VT100

Wikipedia ([https://commons.wikimedia.org/wiki/File:DEC\\_VT100\\_terminal.jpg](https://commons.wikimedia.org/wiki/File:DEC_VT100_terminal.jpg))

Em 1978 a Digital Equipment Corporation (DEC) lançou o terminal VT100, que foi um marco na história da computação. Ele substituiu o mecanismo de impressão eletromecânico dos TTY por um monitor CRT, e **tornou a sessão de trabalho dinâmica**, em tela, permitindo a edição de texto, formulários e maior velocidade de comunicação.

Mais ainda: não era apenas tela e teclado, era "inteligente"!

Ele conseguia interpretar um conjunto de **códigos de controle** (combinações especiais de caracteres que, quando recebidas pelo terminal, não são exibidos na tela mas interpretados como comandos). Esses códigos de controle permitiam que um programa rodando no mainframe controlasse com precisão a aparência da tela do terminal: mover o cursor, alterar atributos do texto (negrito, sublinhado, etc.), desenhar formulários e tabelas. Fez tanto sucesso que popularizou o padrão ANSI X3.64 (para seqüências de controle) e se tornou o padrão de terminal de fato da computação.

# Confusão de termos: shell, console, terminal, prompt...



DEC VT220

Wikipedia ([https://commons.wikimedia.org/wiki/File:DEC\\_vt220.jpg](https://commons.wikimedia.org/wiki/File:DEC_vt220.jpg))

O usuário do terminal era o consumidor final dos serviços do mainframe: o programador, o estudante, o contador, etc.

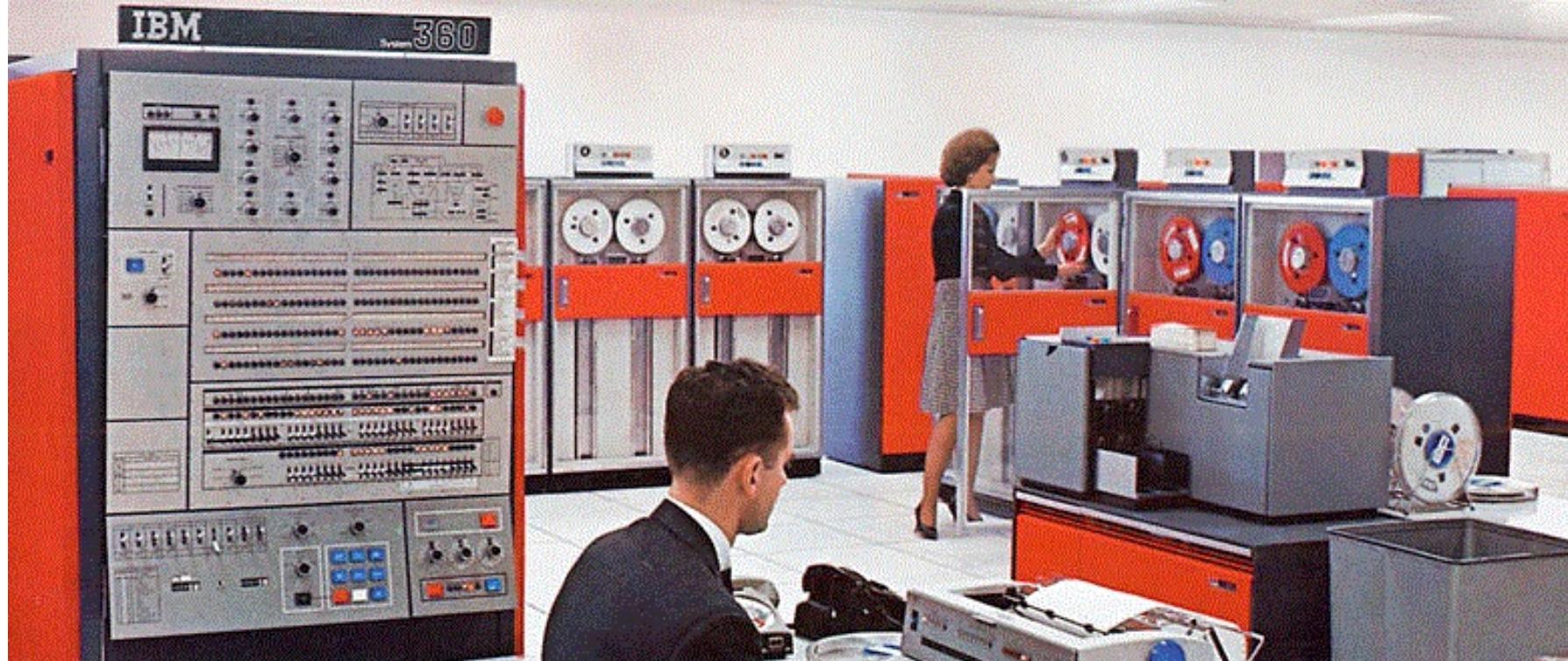
Esses usuários usavam o terminal e, ao digitar os comandos no terminal, o terminal enviava os comandos ao shell, que interpretava e iniciava a seqüência de execução dos comandos.

O terminal não tinha acesso direto a nenhum recurso do hardware, precisava do shell para isso.

# Confusão de termos: shell, console, terminal, prompt...

Característica	Console	Terminal (TTY, VT100, etc.)
Função	Controle, monitoramento e manutenção do sistema	Interação com aplicações, entrada de dados, programação
Privilégio	Absoluto, com acesso direto ao hardware, contornando o S.O.	Restrito, dentro de uma sessão e "vigiado" pelo shell
Usuário típico	Administrador de sistema	Programador, analista, usuário final
Conexão física	Diretamente integrado ou conectado ao complexo da CPU	Remoto, conectado via linhas seriais, controladores ou modens
Tipo de interface	Interruptores, seletores, botões, luzes, etc.	Estilo máquina de escrever (TTY) ou baseado em tela (VT100)
Papel conceitual	Painel de controle da máquina	Janela de acesso ao sistema
Número por sistema	Em geral apenas 1 console	Dezenas ou centenas

# Confusão de termos: shell, console, terminal, prompt...



IBM System/360  
<https://personal.garrettfuller.org/blog/2017/10/28/ibm-system360/>

Existia uma separação muito clara do console para a administração, e dos terminais para o uso.



# Confusão de termos: shell, console, terminal, prompt...



IBM PC XT  
Wikipedia ([https://commons.wikimedia.org/wiki/File:IBM\\_PC\\_XT\\_color.jpg](https://commons.wikimedia.org/wiki/File:IBM_PC_XT_color.jpg))

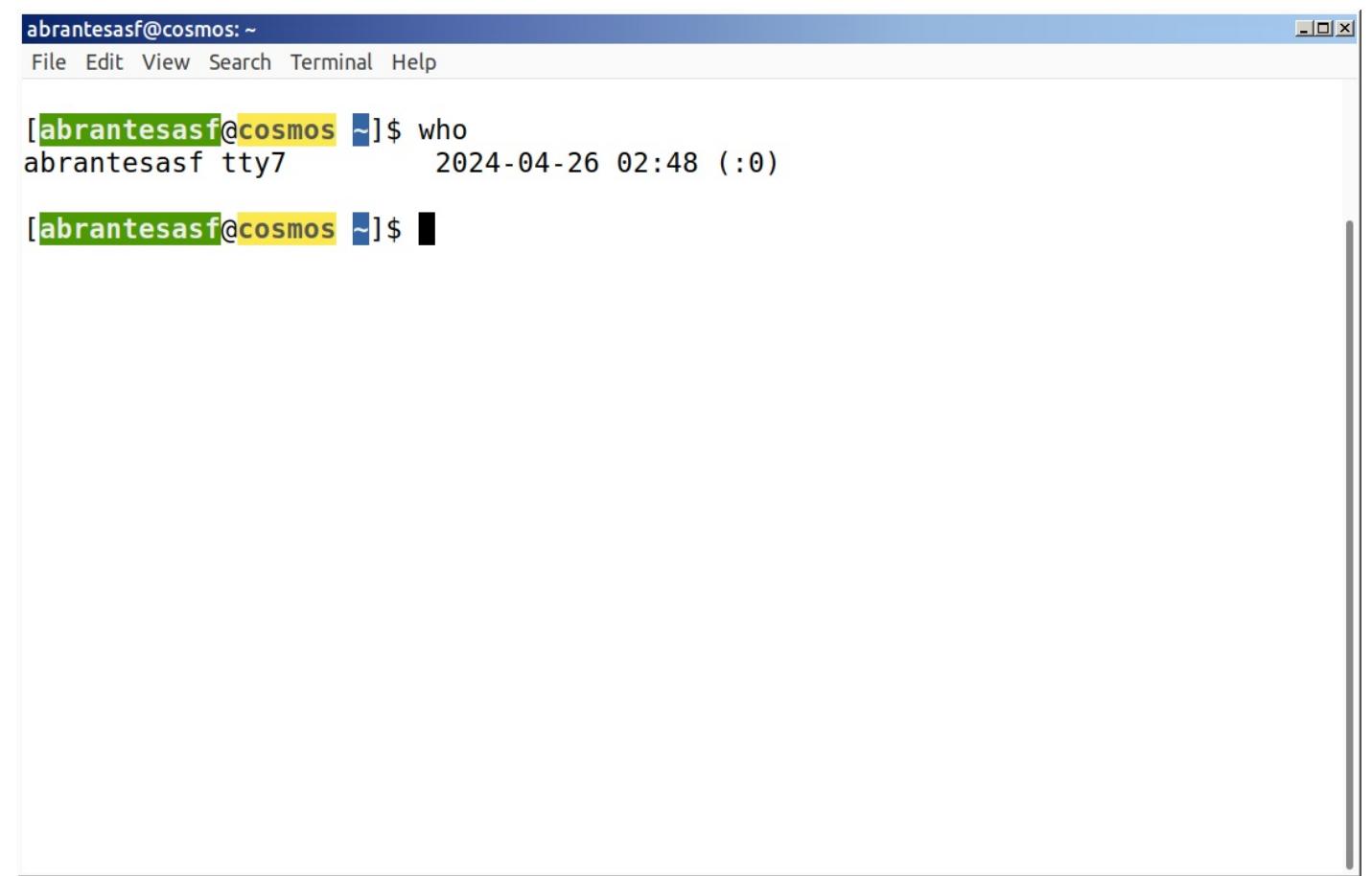
Com a evolução da computação o computador "saiu" da sala do mainframe e foi para a mesa do usuário.

A separação clara entre o console de administração e o terminal do usuário sumiu.

Esses conceitos tiveram que ser reinventados como abstrações de software!

Terminal	→	Emulador de Terminal
Console	→	Console Virtual

# Confusão de termos: shell, console, terminal, prompt...



A screenshot of a terminal window titled "abrantesASF@cosmos: ~". The window includes a menu bar with File, Edit, View, Search, Terminal, and Help. The main area shows the command "[abrantesASF@cosmos ~]\$ who" followed by the output "abrantesASF tty7 2024-04-26 02:48 (:0)". Below this, another command "[abrantesASF@cosmos ~]\$ █" is visible.

Um **EMULADOR DE TERMINAL** é uma aplicação de software que abre uma janela na interface gráfica do usuário e se comporta exatamente como um terminal de hardware clássico como o VT100.

Interpreta as seqüências de escape para **controlar cores, posição do cursor e atributos do texto (texto, destaque, negrito, etc.)**, **esquema de cores, tamanhos, etc.**. Também captura as teclas digitadas e as envia para o shell.

O emulador de terminal fornece uma janela para que o usuário possa interagir com o shell, que roda "dentro" desse terminal emulado. É o shell que exibe o **PROMPT** de comando, lê a linha digitada, executa as ações solicitadas e faz as chamadas de sistema necessárias, e retorna a saída de volta para o emulador de terminal que exiba na tela.

# Confusão de termos: shell, console, terminal, prompt...



```
Ubuntu 24.04.2 LTS server01 tty1
server01 login: abrantesasf
Password:
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-71-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Tue Aug  5 06:39:39 AM UTC 2025

System load: 1.87           Processes:          245
Usage of /: 38.8% of 13.67GB Users logged in:      0
Memory usage: 7%
Swap usage:  0%             IPv4 address for ens3: 172.16.160.131

Expanded Security Maintenance for Applications is not enabled.

81 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

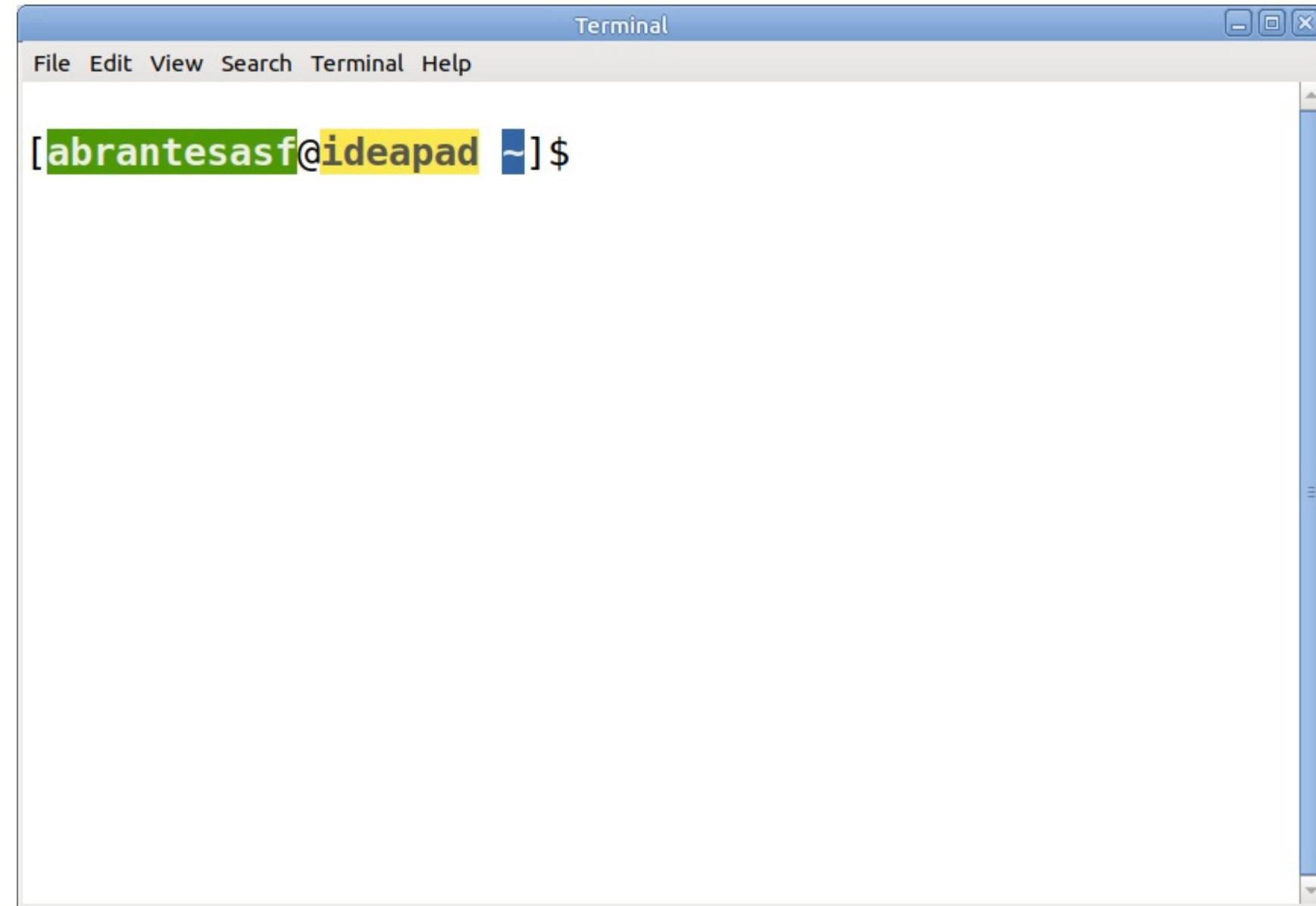
abrantesasf@server01:~$
```

Um **CONSOLE VIRTUAL** é uma aplicação administrativa integrada diretamente no kernel do sistema operacional, em modo texto, que opera de modo independente da interface gráfica do usuário (que pode nem estar instalada). No UNIX/Linux são fornecidos, em geral, 6 consoles virtuais (Ctrl-Alt-F1 até F6).

Para confundir, também são chamados de **terminais virtuais** (não confundir com emulador de terminal).

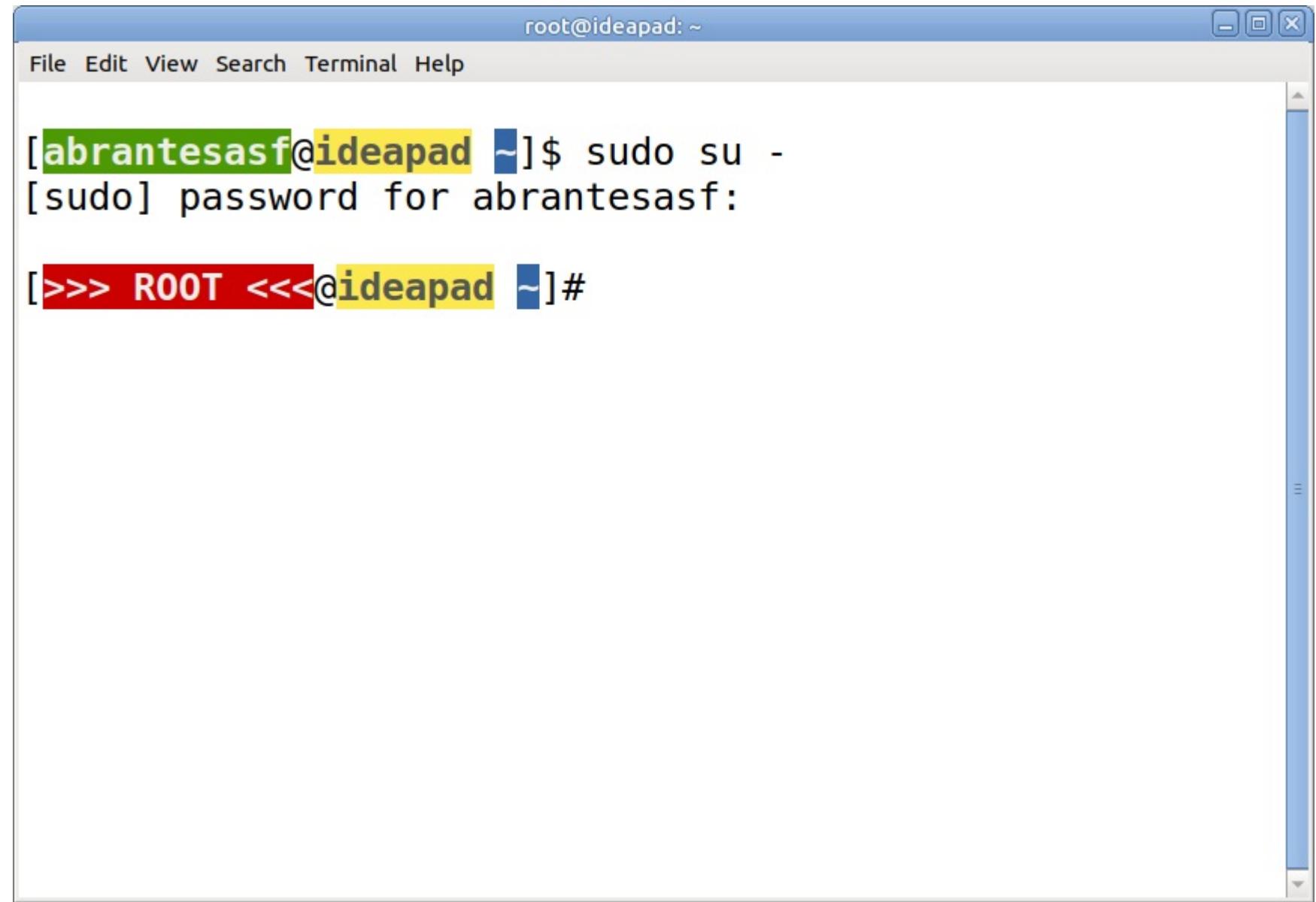
O console virtual faz o papel do console clássico: uma interface de baixo nível, robusta e confiável para a administração do sistema, solução de problemas e recuperação de desastres, especialmente se a GUI falhar ou não estiver instalada.

# Confusão de termos: shell, console, terminal, prompt...



Emulador de terminal sendo usado como terminal de usuário!

# Confusão de termos: shell, console, terminal, prompt...



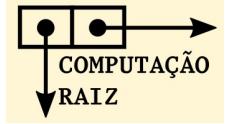
The screenshot shows a terminal window titled "root@ideapad: ~". The window has a standard window title bar with icons for minimize, maximize, and close. Below the title bar is a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The main area of the terminal displays the following text:

```
[abrantesasf@ideapad ~]$ sudo su -
[sudo] password for abrantesasf:

[>>> ROOT <<<@ideapad ~]#
```

Emulador de terminal sendo usado como console administrativo!

# Confusão de termos: shell, console, terminal, prompt...



```
Ubuntu 24.04.2 LTS server01 tty1

server01 login: abrantesASF
Password:
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-71-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

System information as of Tue Aug  5 06:51:18 AM UTC 2025

System load: 1.43          Processes:      245
Usage of /: 38.8% of 13.67GB  Users logged in:   0
Memory usage: 7%           IPv4 address for ens33: 172.16.160.131
Swap usage:  0%

Expanded Security Maintenance for Applications is not enabled.

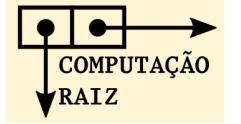
81 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

abrantesASF@server01:~$ _
```

**Console virtual sendo usado como terminal de usuário!**

# Confusão de termos: shell, console, terminal, prompt...



```
Ubuntu 24.04.2 LTS server01 tty1

server01 login: abrantesASF
Password:
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-71-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Tue Aug  5 06:51:18 AM UTC 2025

System load: 1.43          Processes:           245
Usage of /: 38.8% of 13.67GB  Users logged in:      0
Memory usage: 7%            IPv4 address for ens33: 172.16.160.131
Swap usage:  0%

Expanded Security Maintenance for Applications is not enabled.

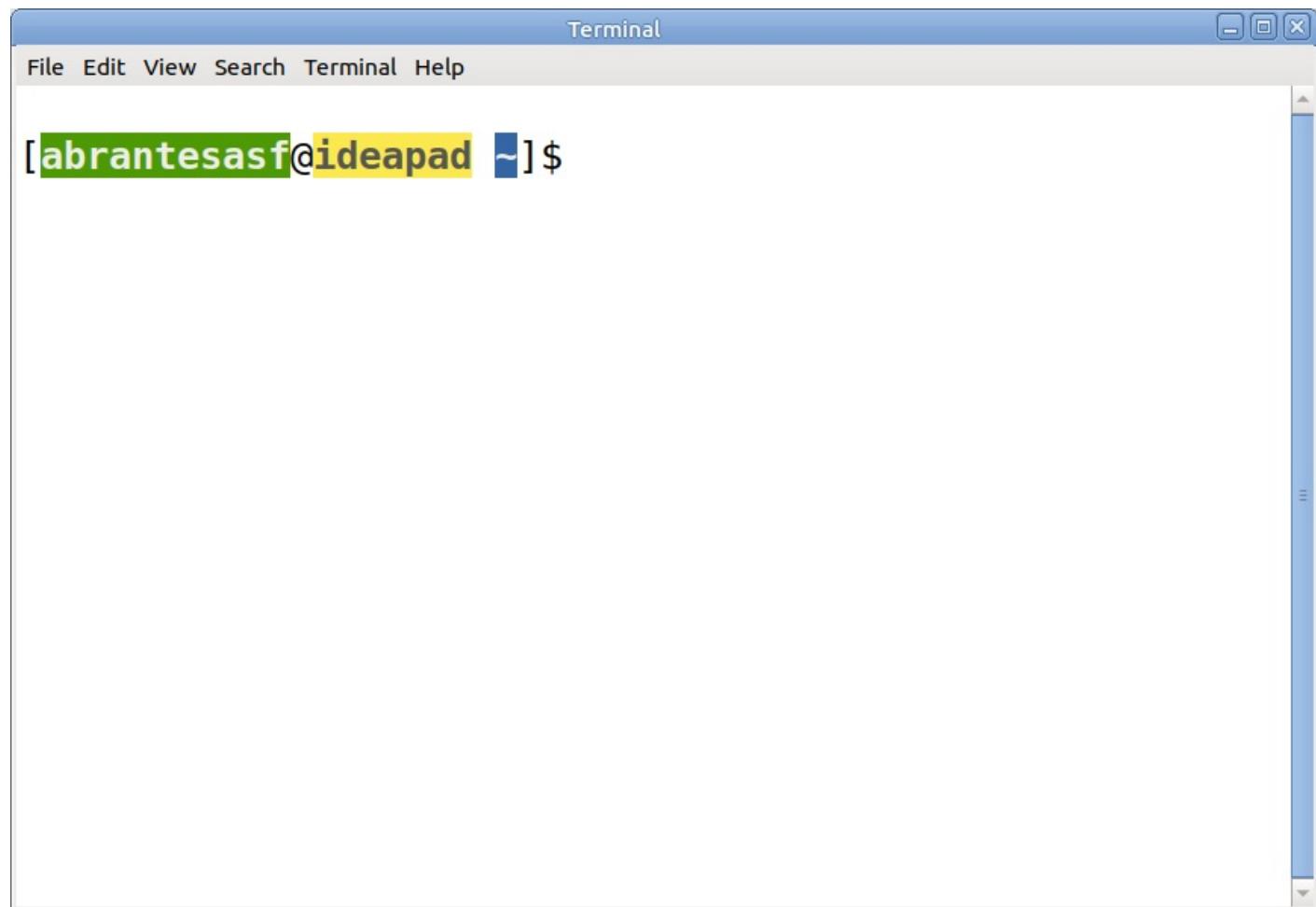
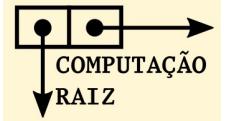
81 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

abrantesASF@server01:~$ sudo su -
[sudo] password for abrantesASF:
root@server01:~# _
```

**Console virtual sendo usado como console administrativo!**

# Confusão de termos: shell, console, terminal, prompt...



O UNIX/Linux é tão poderoso que você pode escolher absolutamente tudo:

- a) Que emulador de terminal você quer?
- b) Que shell você quer?
- c) Que interface gráfica você quer?
- d) ...

Mate Terminal rodando bash!

# Confusão de termos: shell, console, terminal, prompt...

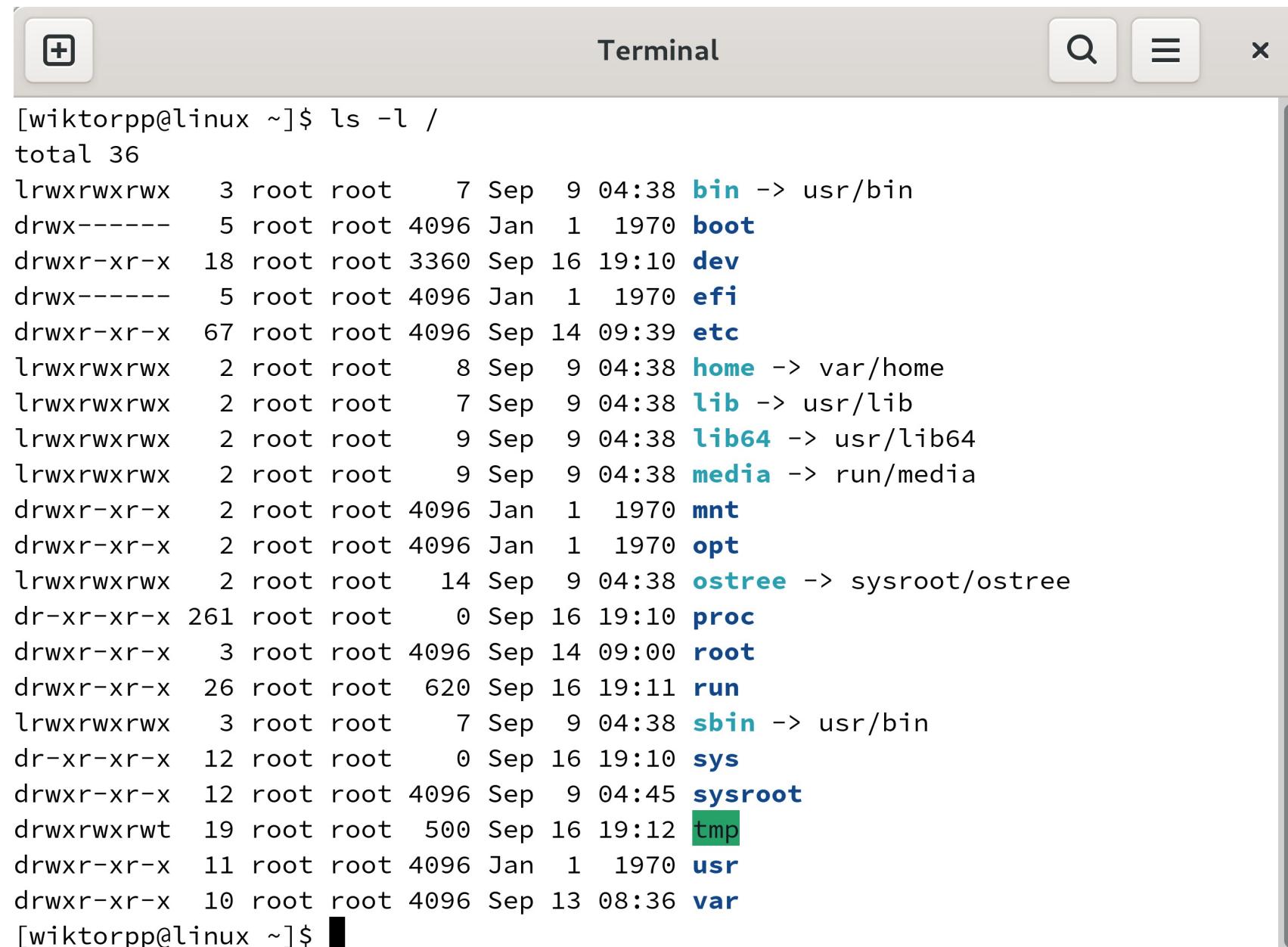
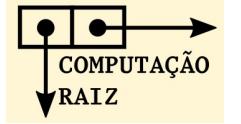
```

[~/home]$ ls
vidarlo
[~/home]$ cd ..
[~/]$ cd etc
[~/etc]$ ls
0.0.10.in-addr.arpa    csh.cshrc      gshadow-        logrotate.d      odbcinst.ini   rmt
adduser.conf            csh.login       gtk             lynx.cfg        openoffice     rpc
adjtime                 csh.logout      host.conf       magic           opt
aliases                 db.cache       hostname        mailcap         pam.conf      screenrc
alternatives            debconf.conf   hosts          mailcap.order  pam.d        security
apm                     debian_version hosts.allow     mailname        passwd       services
apt                     default        hosts.deny     mail.rc         passwd-      shadow
asterisk                defoma        hotplug        manpath.config perl
at.deny                 deluser.conf   hotplug.d      mdadm          PPP
bakipkungfu             dhclient.conf  identd.conf   mediaprm       printcap     shells
bash.bashrc              dhclient-script identd.key     mime.types     profile
bash_completion          dictionaries-common inetc.conf   mkinitrd       protocols   squid
bash_completion.d        discover.conf  init.d        modprobe.d    python2.3
bind                    discover.conf-2.6 inittab       modules       raidtab
blkid.tab               discover.d     inputrc       modules.conf  rc0.d        syslog.conf
blkid.tab.old            dpkg          ipkungfu     modules.conf.old rc1.d        terminfo
calendar                emacs          issue         modutils      rc2.d        timezone
chatscripts              emacs21        issue.net     motd          rc3.d        ucf.conf
chkrootkit.conf          email-addresses kernel-img.conf mtab          rc4.d        updatedb.conf
complete.tcsh             environment   ldap          mtools.conf   rc5.d        vidarlo.net.hosts
console                 exim4          ld.so.cache   Muttrc         rc6.d        w3m
console-tools            fdmount.conf  ld.so.conf    mysql          rc.d         wgetrc
cron.d                  fonts          locale.alias  nanorc         rcS.d        #wvdial.conf#
cron.daily               fstab          locale.gen    network        reportbug.conf
cron.hourly              groff          localtime   networks       resolvconf   wvdial.conf
cron.monthly              group          logcheck     nsswitch.conf  resolv.conf  wvdial.conf~
cron.monthly              group-         login.defs   ODBCDataSources resolv.conf~ X11
cron.weekly              gshadow        logrotate.conf odbc.ini      resolv.conf  xpilot
[~/etc]$ []

```

Xterm rodando bash!

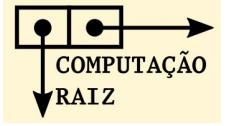
# Confusão de termos: shell, console, terminal, prompt...



```
[wiktorpp@linux ~]$ ls -l /
total 36
lrwxrwxrwx  3 root root   7 Sep  9 04:38 bin  -> usr/bin
drwx-----  5 root root 4096 Jan  1 1970 boot
drwxr-xr-x 18 root root 3360 Sep 16 19:10 dev
drwx-----  5 root root 4096 Jan  1 1970 efi
drwxr-xr-x 67 root root 4096 Sep 14 09:39 etc
lrwxrwxrwx  2 root root   8 Sep  9 04:38 home -> var/home
lrwxrwxrwx  2 root root   7 Sep  9 04:38 lib   -> usr/lib
lrwxrwxrwx  2 root root   9 Sep  9 04:38 lib64 -> usr/lib64
lrwxrwxrwx  2 root root   9 Sep  9 04:38 media -> run/media
drwxr-xr-x  2 root root 4096 Jan  1 1970 mnt
drwxr-xr-x  2 root root 4096 Jan  1 1970 opt
lrwxrwxrwx  2 root root  14 Sep  9 04:38 ostree -> sysroot/ostree
dr-xr-xr-x 261 root root   0 Sep 16 19:10 proc
drwxr-xr-x  3 root root 4096 Sep 14 09:00 root
drwxr-xr-x 26 root root  620 Sep 16 19:11 run
lrwxrwxrwx  3 root root   7 Sep  9 04:38 sbin -> usr/bin
dr-xr-xr-x 12 root root   0 Sep 16 19:10 sys
drwxr-xr-x 12 root root 4096 Sep  9 04:45 sysroot
drwxrwxrwt 19 root root  500 Sep 16 19:12 tmp
drwxr-xr-x 11 root root 4096 Jan  1 1970 usr
drwxr-xr-x 10 root root 4096 Sep 13 08:36 var
[wiktorpp@linux ~]$ █
```

Gnome Terminal rodando bash!

# Confusão de termos: shell, console, terminal, prompt...

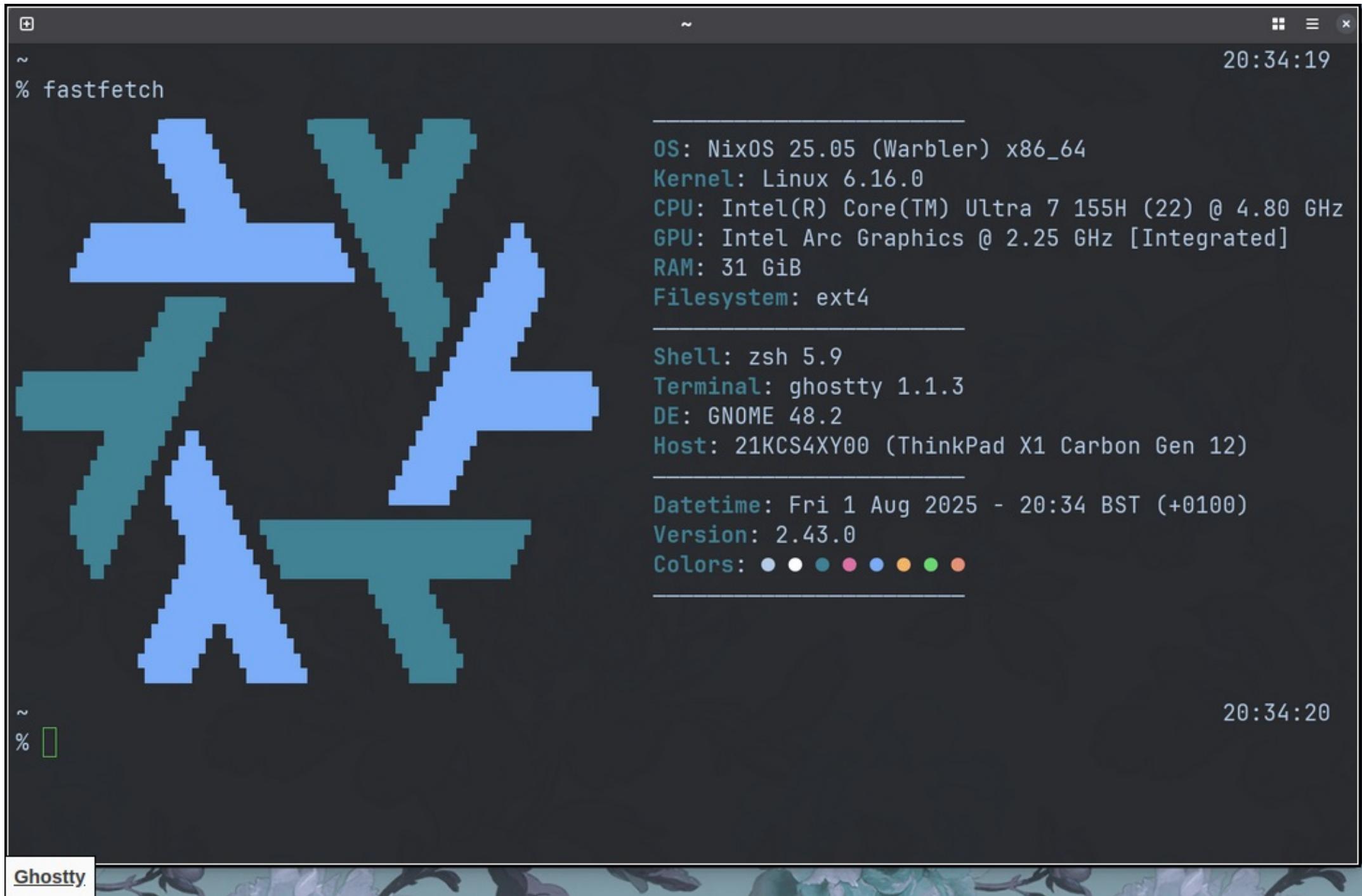


```
Honkai Star Rail - Nameless Faces : zsh — Konsole
New Tab Split View Copy Paste Find... ▾
[Mon 25/01/13 15:29 WIB] [pts/1] [x86_64/linux-gnu/6.12.8-2-default] [5.9]
<vulcansphere@aventurine-topaz:~/Music/Honkai Star Rail - Nameless Faces>
zsh 809 % ls -l
total 168696
-rw-r--r-- 1 vulcansphere vulcansphere 33754619 Jan 11 10:36 '01. Nameless Faces (feat. Lilas Ikuta) [英語版].flac'
-rw-r--r-- 1 vulcansphere vulcansphere 33674233 Jan 11 10:36 '02. Nameless Faces (feat. 谭晶) [中国語版].flac'
-rw-r--r-- 1 vulcansphere vulcansphere 31357011 Jan 11 10:36 '03. Nameless Faces (Instrumental).flac'
-rw-r--r-- 1 vulcansphere vulcansphere 31883318 Jan 11 10:36 '04. Nameless Faces (英語版オフボーカル音源).flac'
-rw-r--r-- 1 vulcansphere vulcansphere 32012296 Jan 11 10:36 '05. Nameless Faces (中国語版オフボーカル音源).flac'
-rw-r--r-- 1 vulcansphere vulcansphere 10048882 Jan 11 10:36 Cover.jpg
[Mon 25/01/13 15:29 WIB] [pts/1] [x86_64/linux-gnu/6.12.8-2-default] [5.9]
<vulcansphere@aventurine-topaz:~/Music/Honkai Star Rail - Nameless Faces>
zsh 810 % ls -l /home/vulcansphere/Music/Genshin\ Impact\ -\ Blazing\ Heart
total 196616
-rw-r--r-- 1 vulcansphere vulcansphere 39868067 Jan 11 11:37 '01. Blazing Heart (Chinese Ver.).flac'
-rw-r--r-- 1 vulcansphere vulcansphere 40029989 Jan 11 11:37 '02. Blazing Heart (English Ver.).flac'
-rw-r--r-- 1 vulcansphere vulcansphere 39719598 Jan 11 11:37 '03. Blazing Heart (feat. 小松未可子) [Japanese Ver.].flac
'
-rw-r--r-- 1 vulcansphere vulcansphere 39733322 Jan 11 11:37 '04. Blazing Heart (Korean Ver.).flac'
-rw-r--r-- 1 vulcansphere vulcansphere 38870606 Jan 11 11:37 '05. Blazing Heart (Instrumental).flac'
-rw-r--r-- 1 vulcansphere vulcansphere 3101613 Jan 11 11:37 cover.jpg
[Mon 25/01/13 15:29 WIB] [pts/1] [x86_64/linux-gnu/6.12.8-2-default] [5.9]
<vulcansphere@aventurine-topaz:~/Music/Honkai Star Rail - Nameless Faces>
zsh 811 % zypper info konsole
Loading repository data...
Reading installed packages...

Information for package konsole:
-----
Repository      : Main Repository (OSS)
Name            : konsole
Version         : 24.12.0-1.1
Arch            : x86_64
Vendor          : openSUSE
Installed Size  : 1.1 MiB
Installed       : Yes (automatically)
Status          : up-to-date
Source package  : konsole-24.12.0-1.1.src
Upstream URL   : https://apps.kde.org/konsole
Summary         : KDE Terminal
Description     :
    Konsole is a terminal emulator for the K Desktop Environment.
```

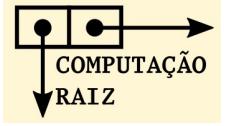
Konsole Terminal rodando zsh!

# Confusão de termos: shell, console, terminal, prompt...



Ghostty Terminal rodando zsh!

# Confusão de termos: shell, console, terminal, prompt...



**Em resumo:**

- Você pode escolher o emulador de terminal que gostar mais
- Você pode escolher o shell que gostar mais

**PADRÃO UNIX/LINUX:**

**Oferecer mecanismos, não políticas!**

# Uso do terminal



O prompt é completamente configurável. Eu gosto assim:

```
[abrantesasf@cosmos ~]$ 
```

```
[>>> ROOT <<<@cosmos ~]# 
```

```
[autolab@autolab ~]$ 
```

```
[>>> ROOT <<<@autolab ~]# 
```