[301] Program I/O

Tyler Caraza-Harter

Learning Objectives

Understand role of operating system in I/O

Use storage

- How do storage drives appear in Windows/UNIX?
- How are files named?

Work from the command line

- What is a terminal emulator?
- What is a shell?

PowerShell and bash

- How to navigate
- Run programs
- Redirect/pipe output
- Save scripts

Today's Topics

Program Input/Output

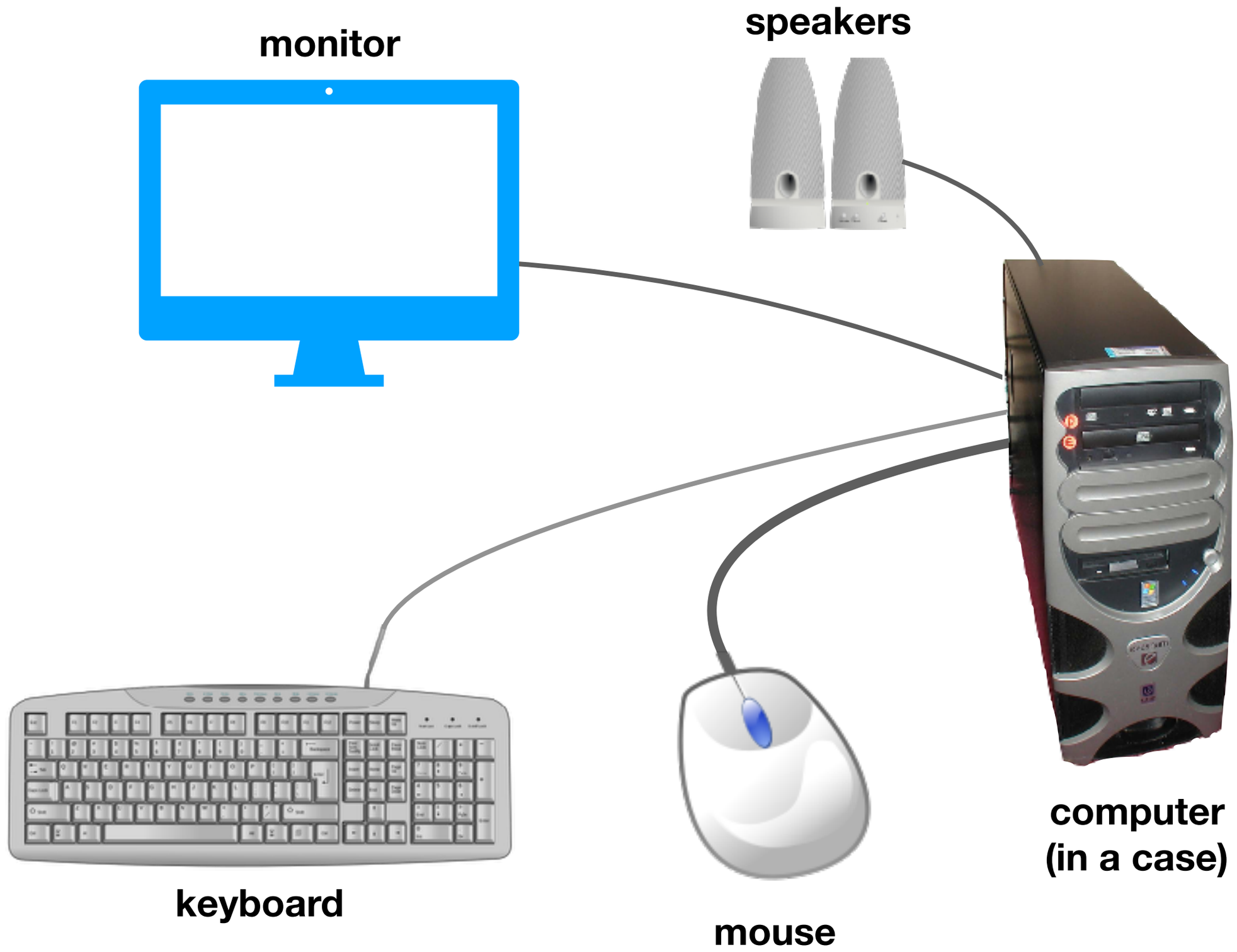
- Review Computer I/O
- Program I/O and the Operating System

File Systems

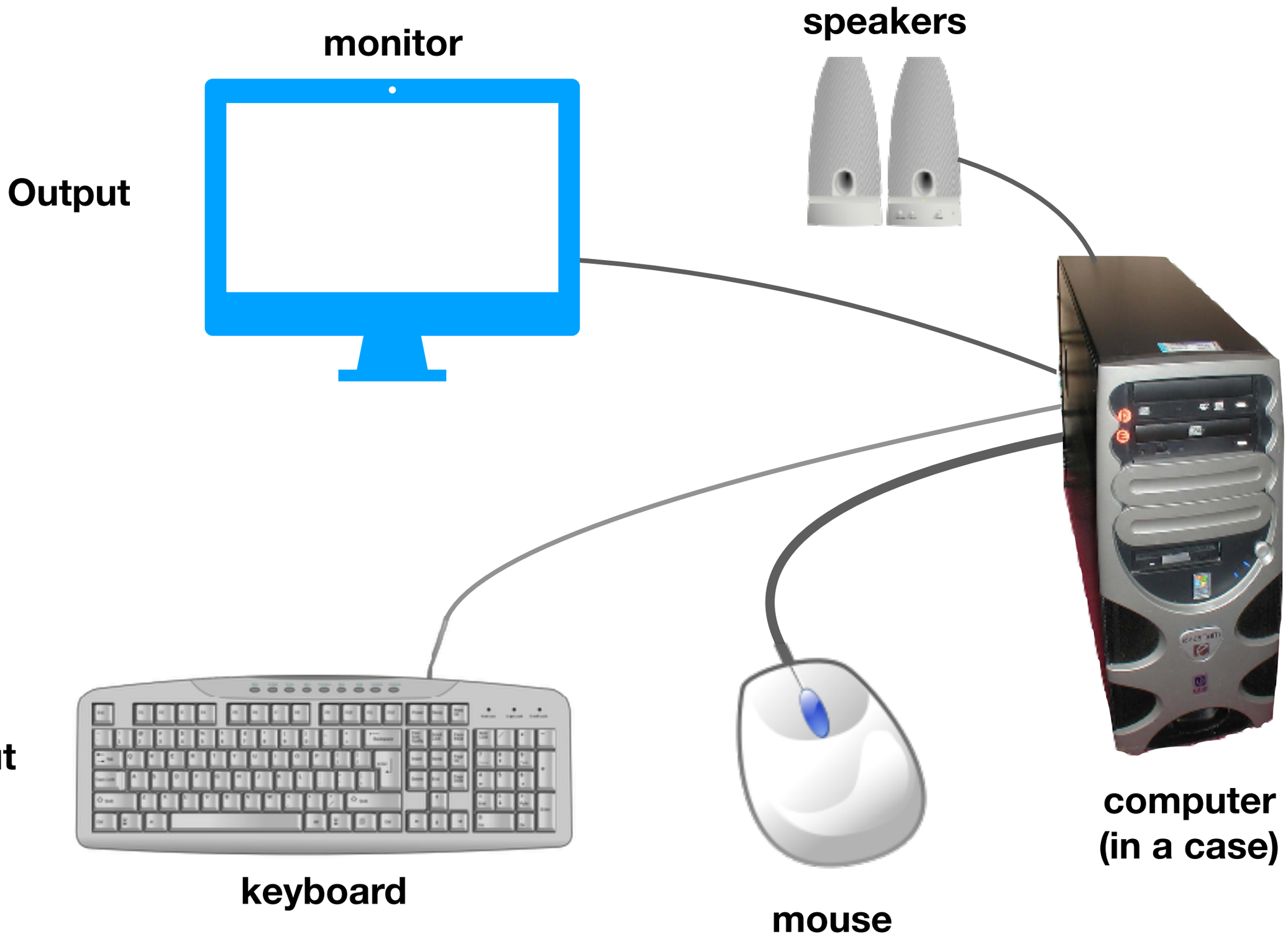
Terminal Emulators and Shells

Tutorials

Computer Input/Output



Computer Input/Output



Computer Input/Output

Are NICs and storage drives input devices or output devices?

computer (in a case)

Today's Topics

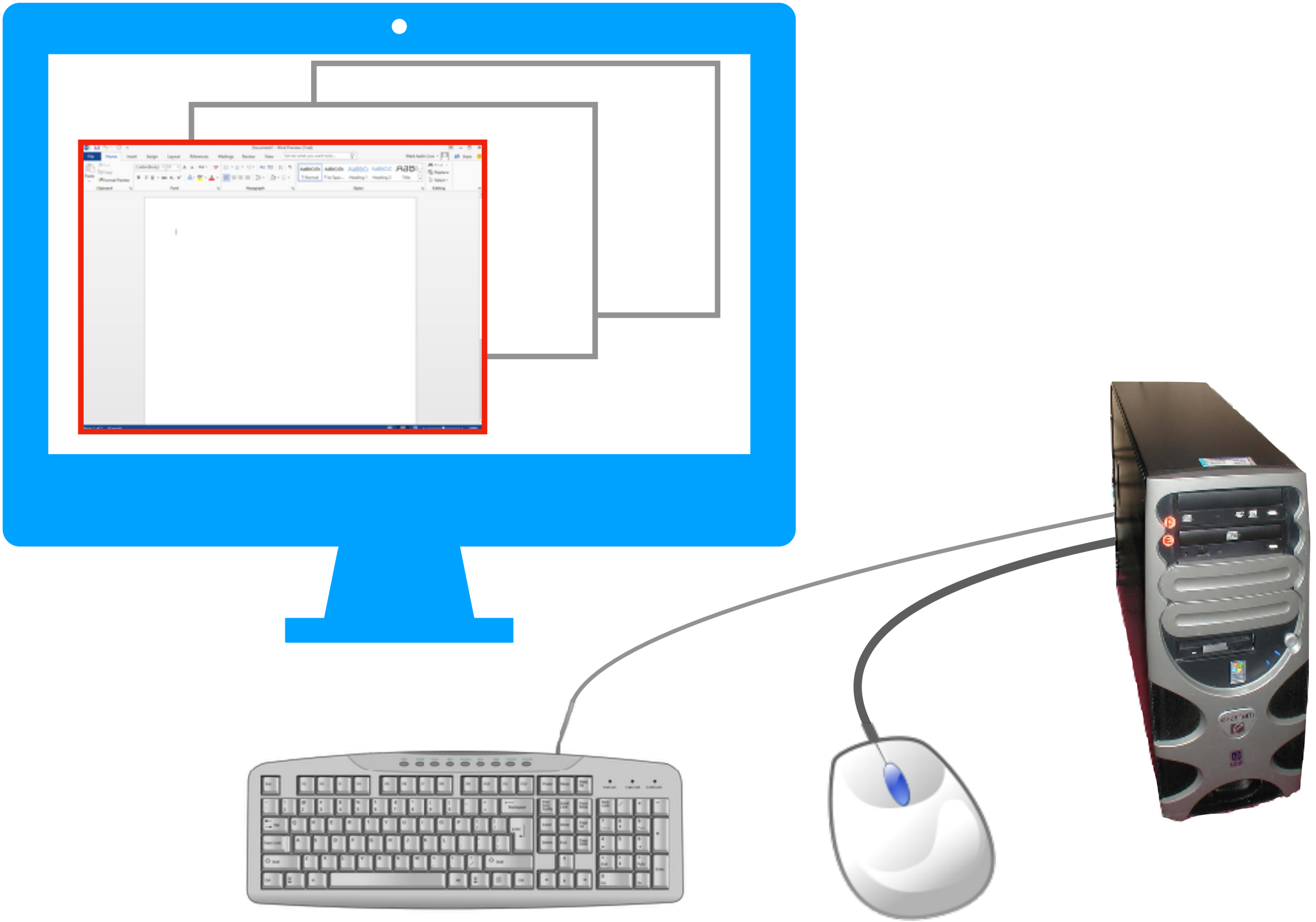
Program Input/Output

- Review Computer I/O
- Program I/O and the Operating System

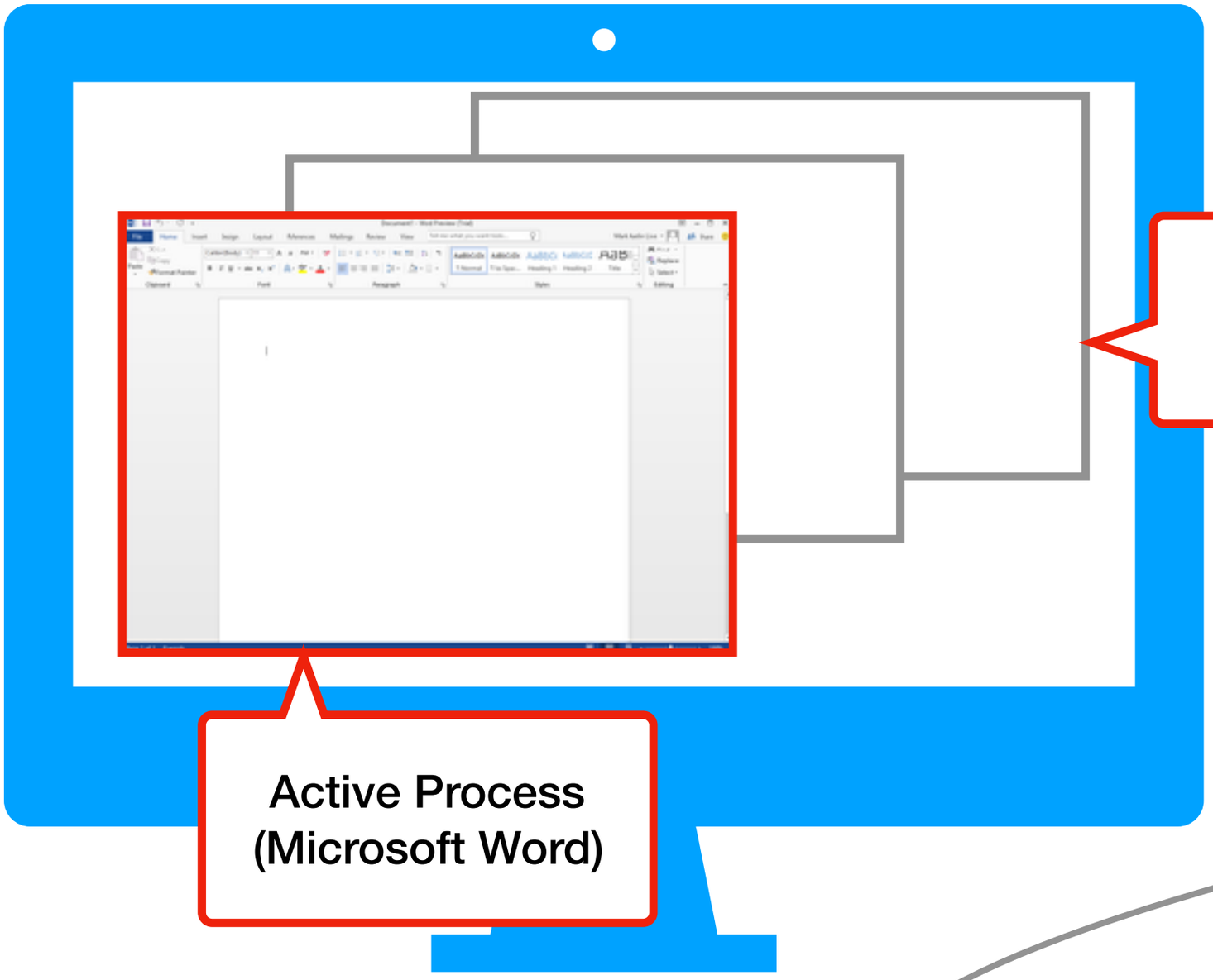
File Systems

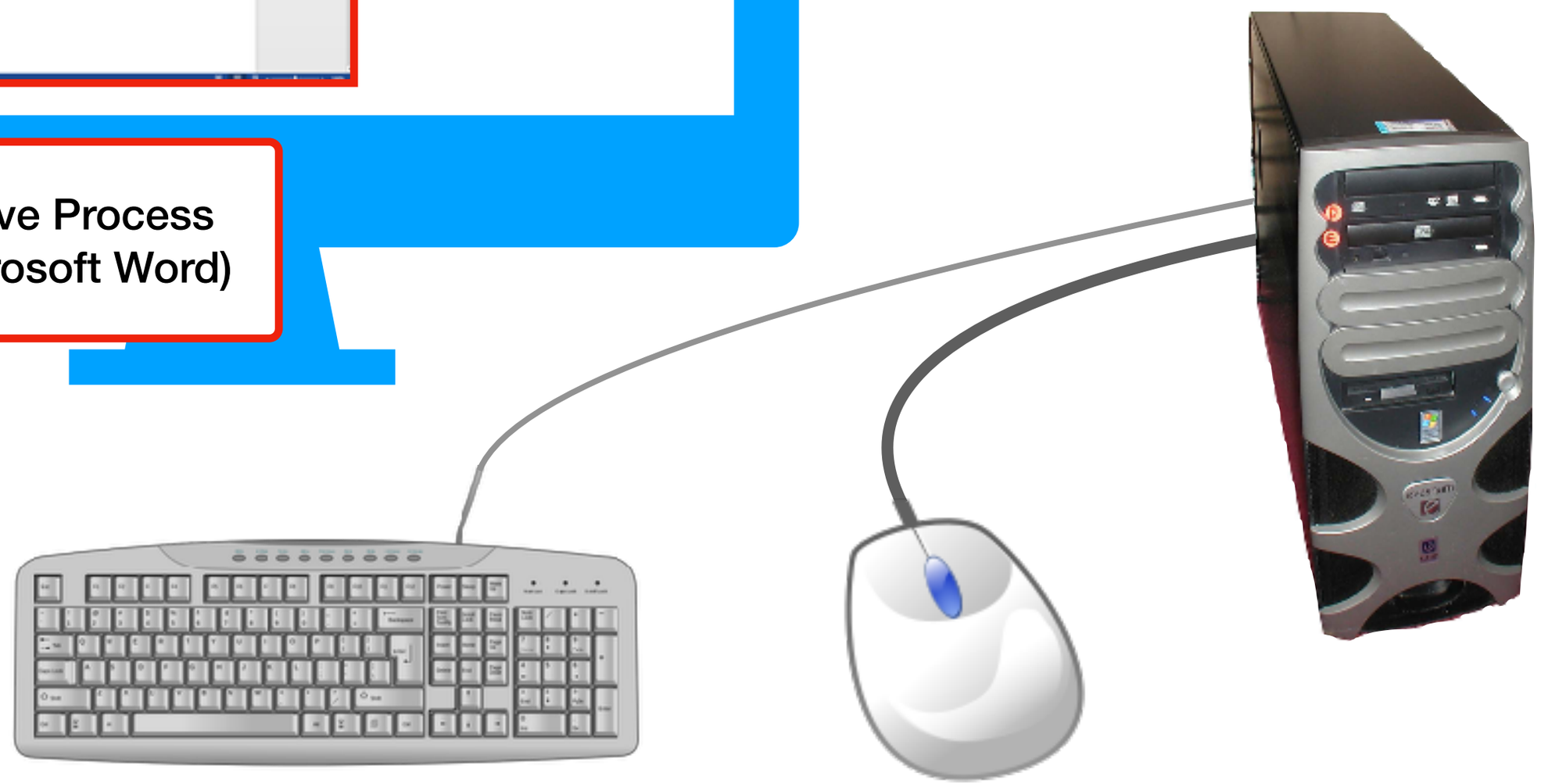
Terminal Emulators and Shells

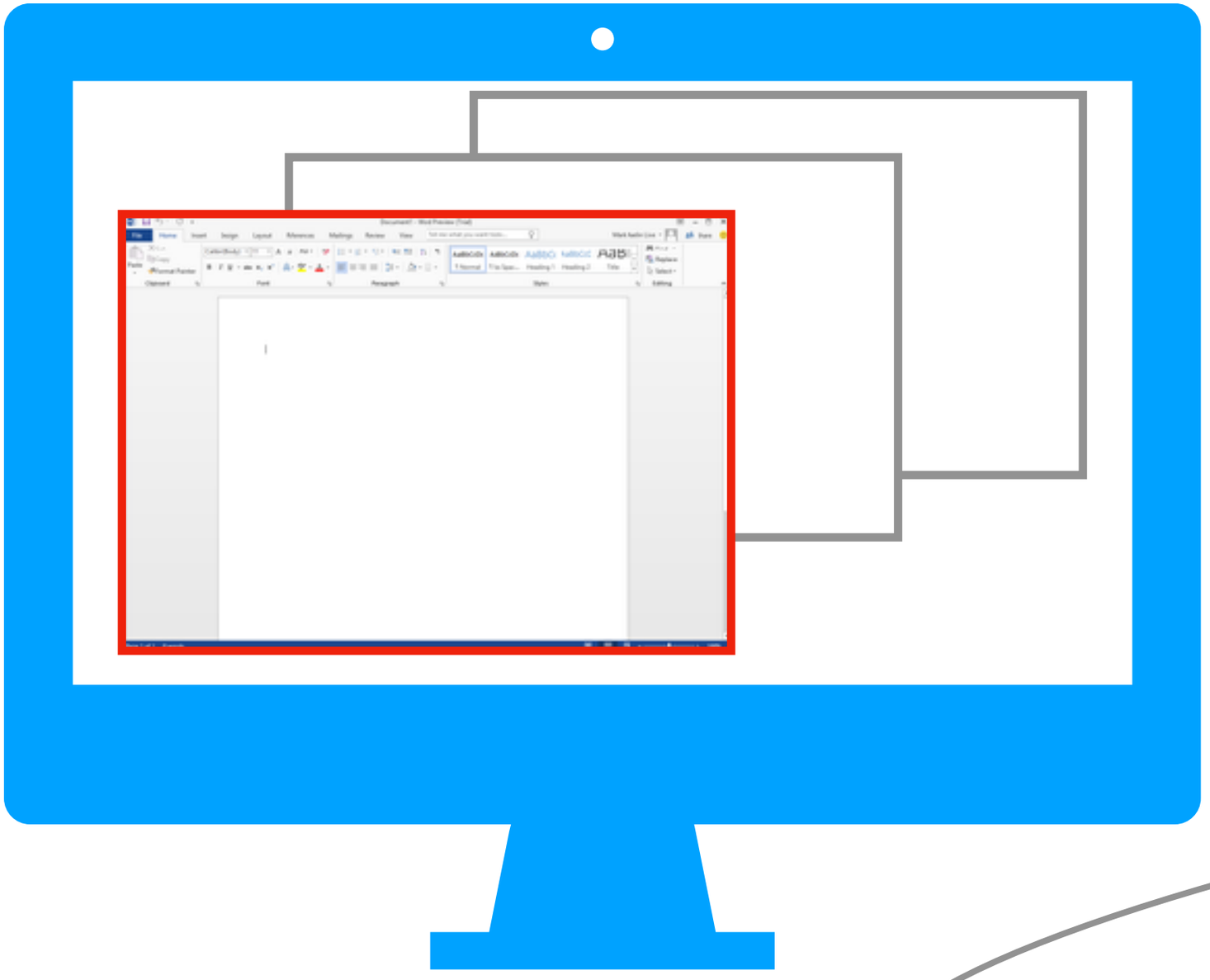
Tutorial



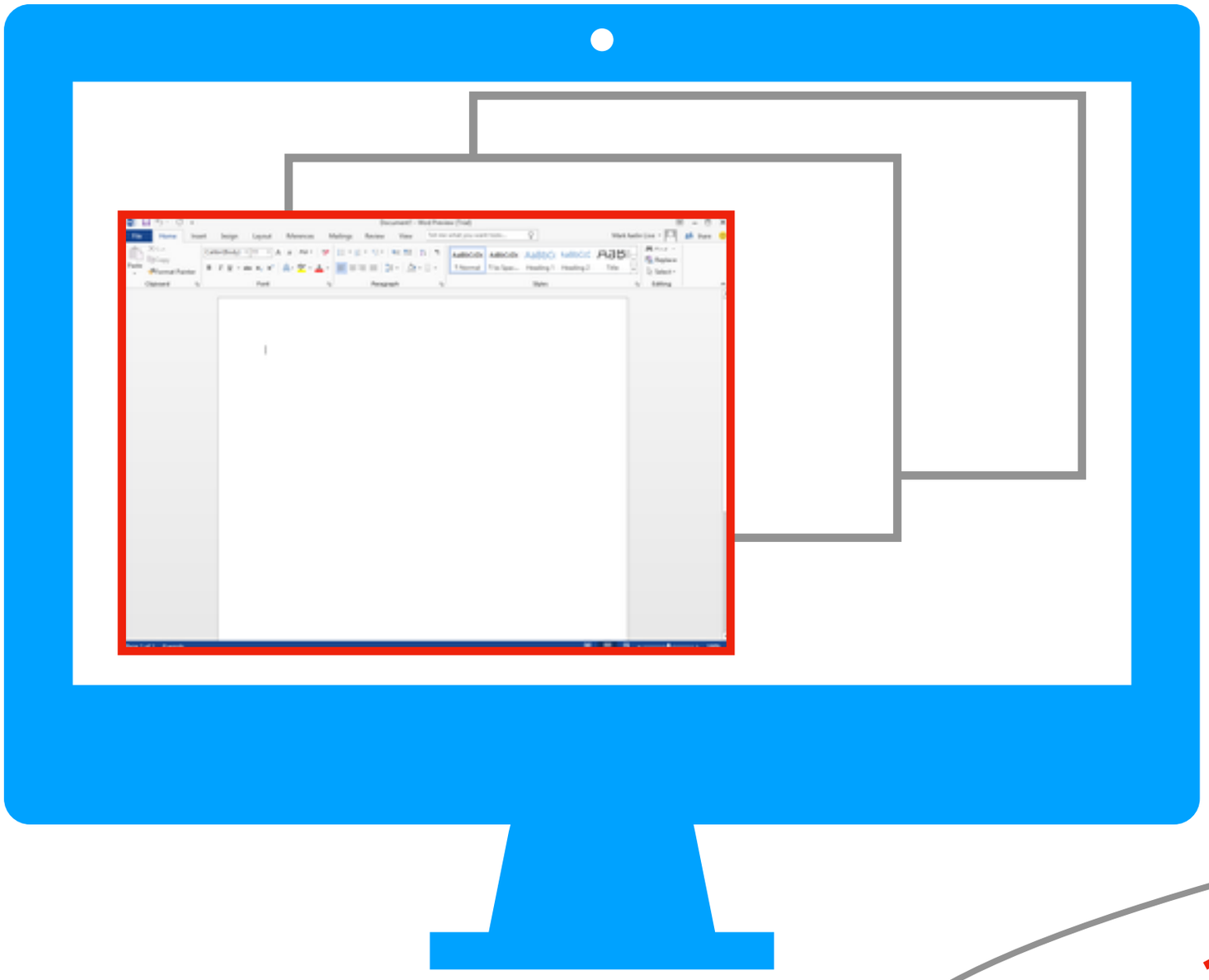
Terminology: a “process” is just a running “program”







Operating System



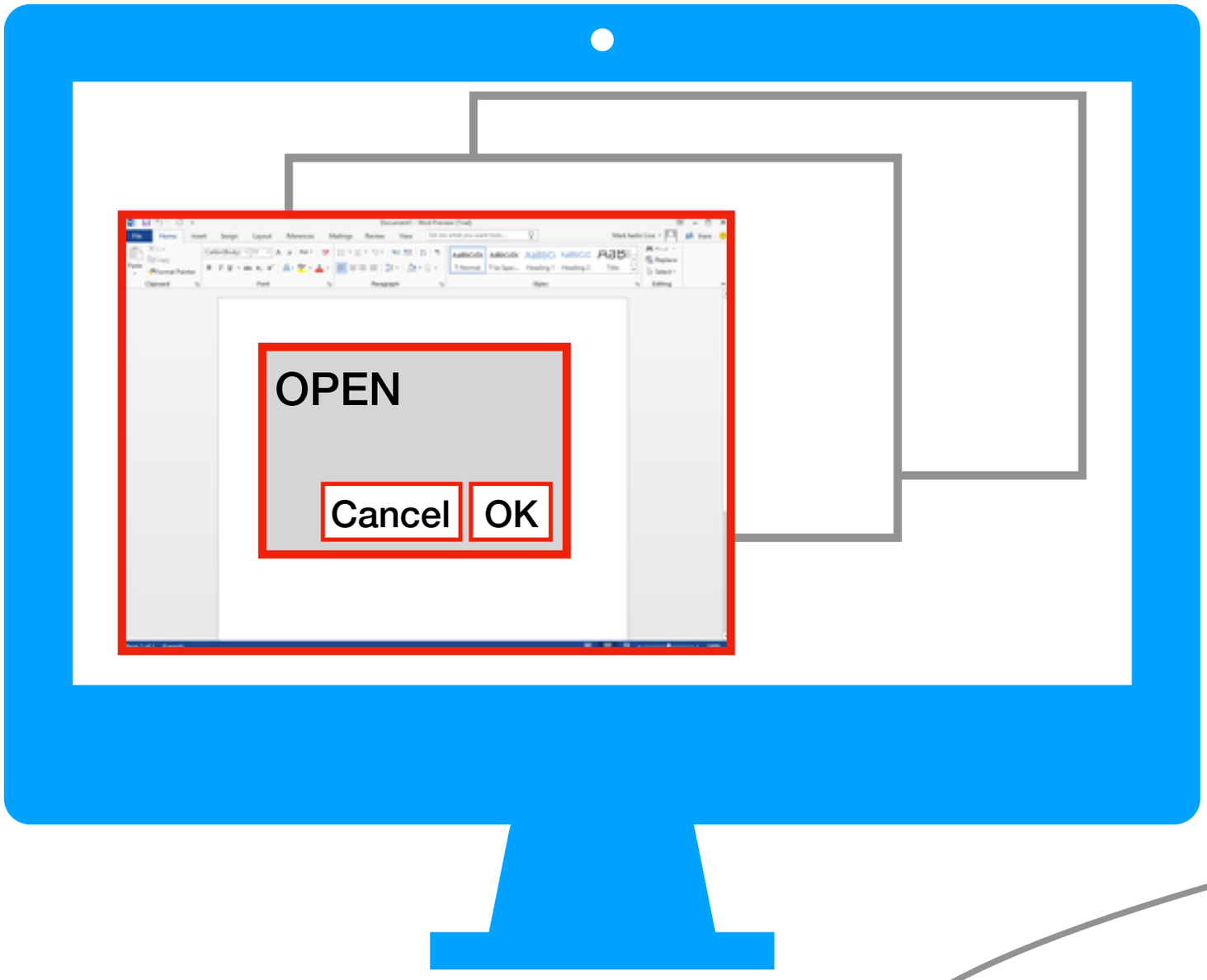
Operating System

Click!
(input)

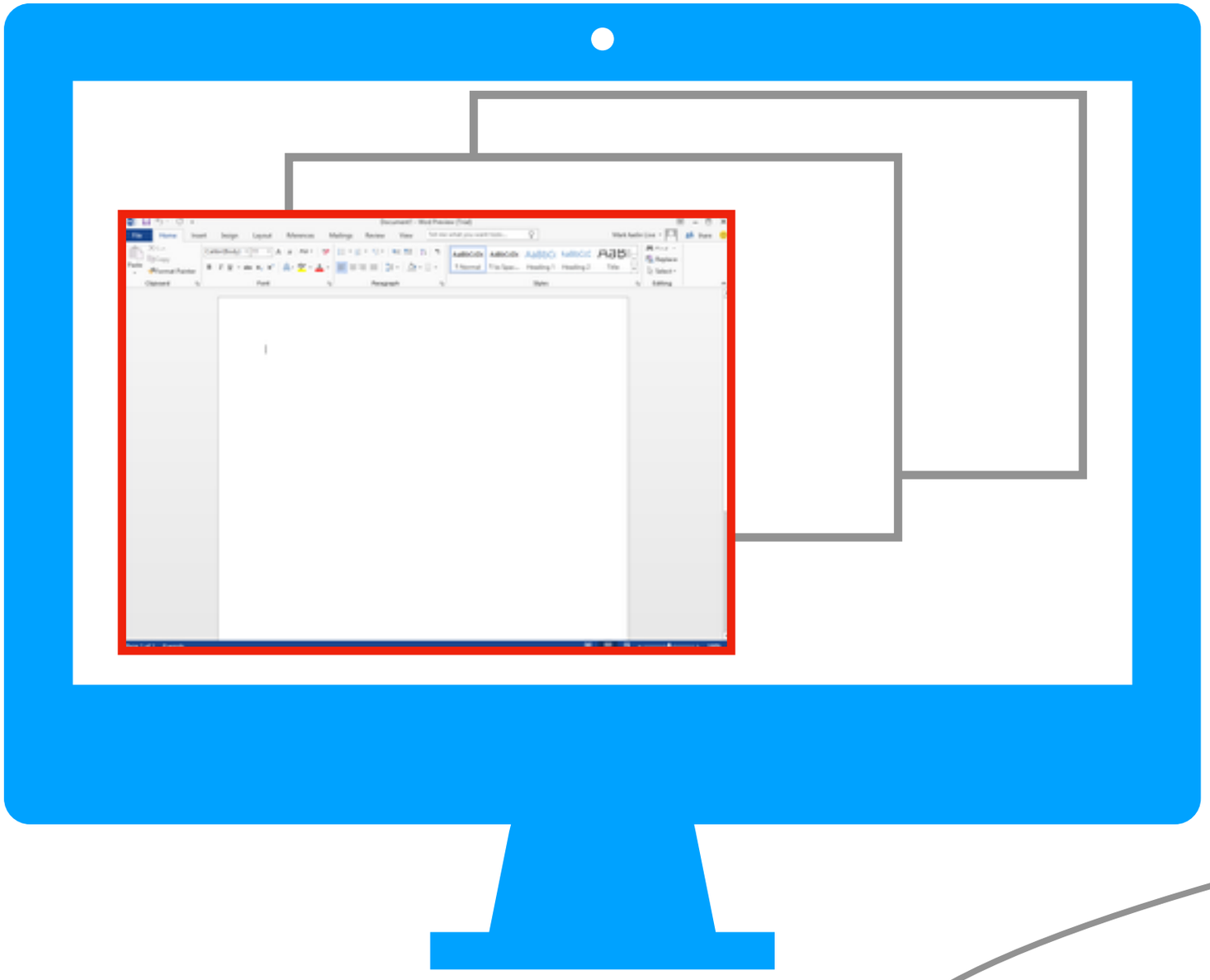
Click!
(input)

Operating System

Click!
(input)

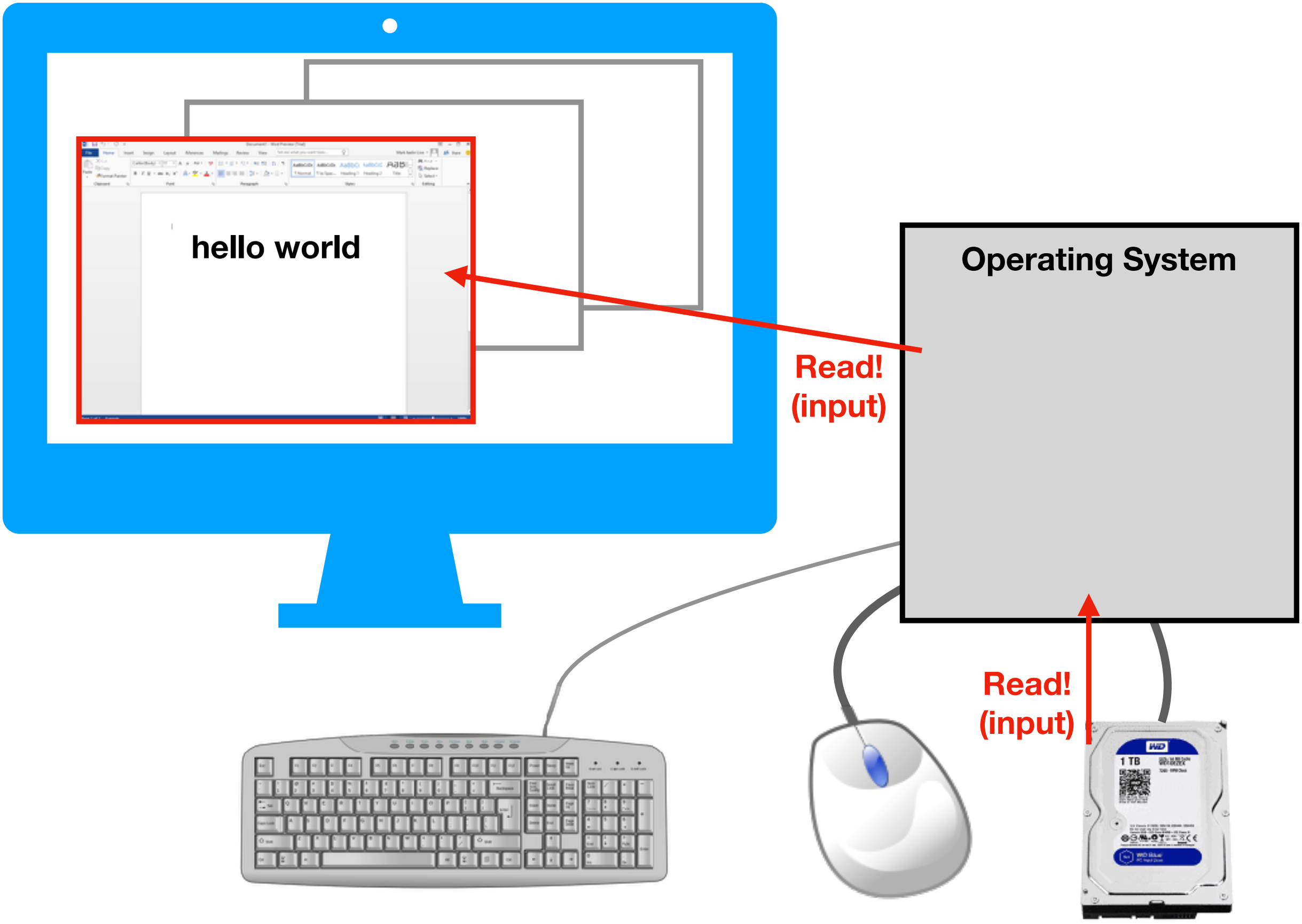


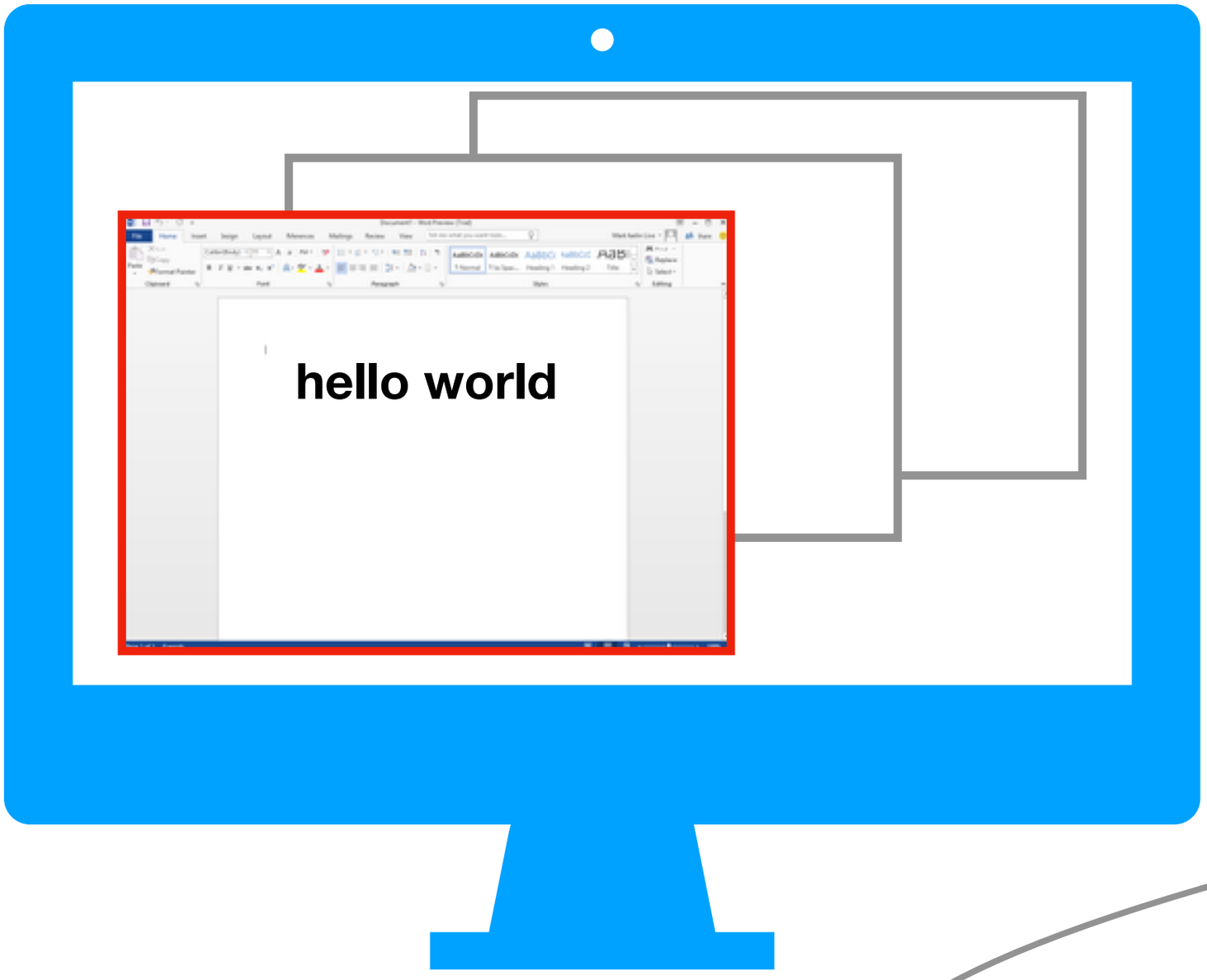
Operating System



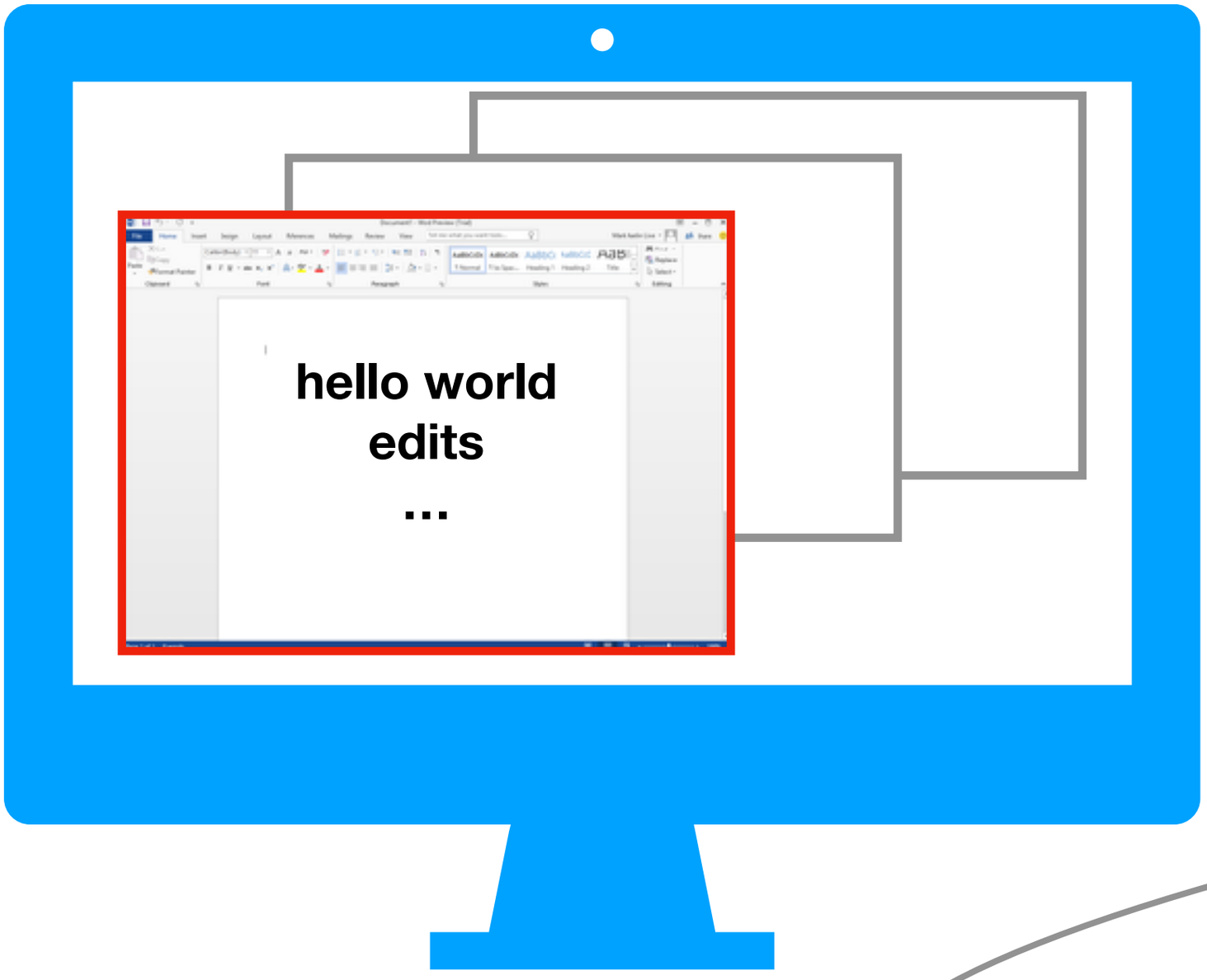
Operating System

Read!
(input)

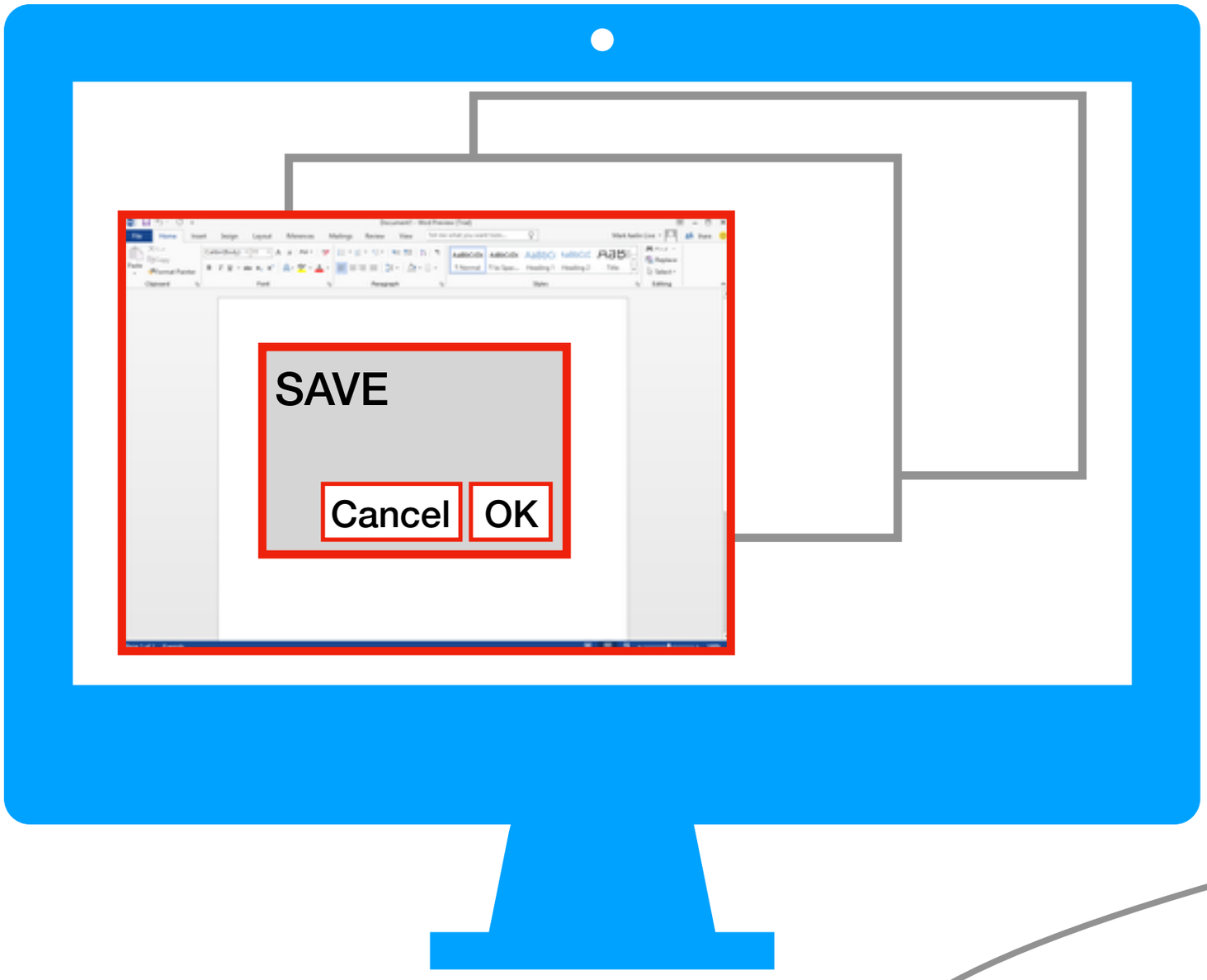

hello world
Read!
(input)
Operating System
Read!
(input)
1 TB



Operating System



Operating System



Operating System

SAVE

Cancel

OK

Write!
(output)

Operating System

1 TB

SAVE

Cancel

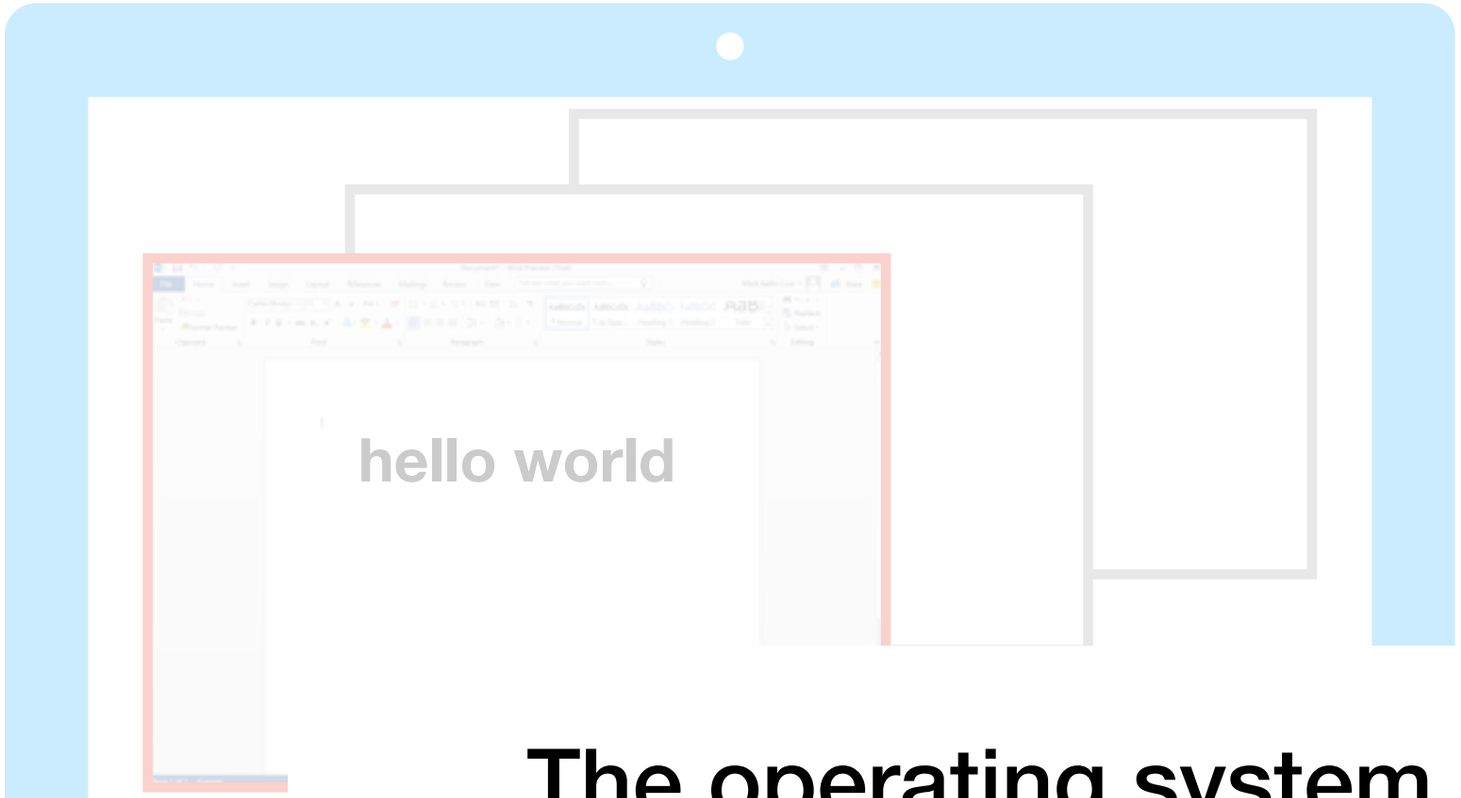
OK

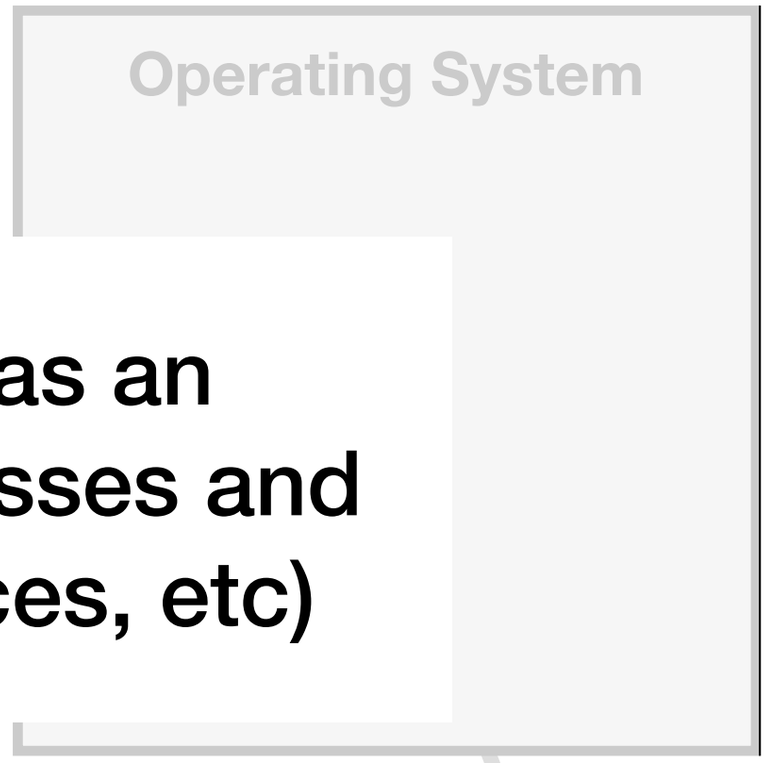
Write!
(output)

Operating System

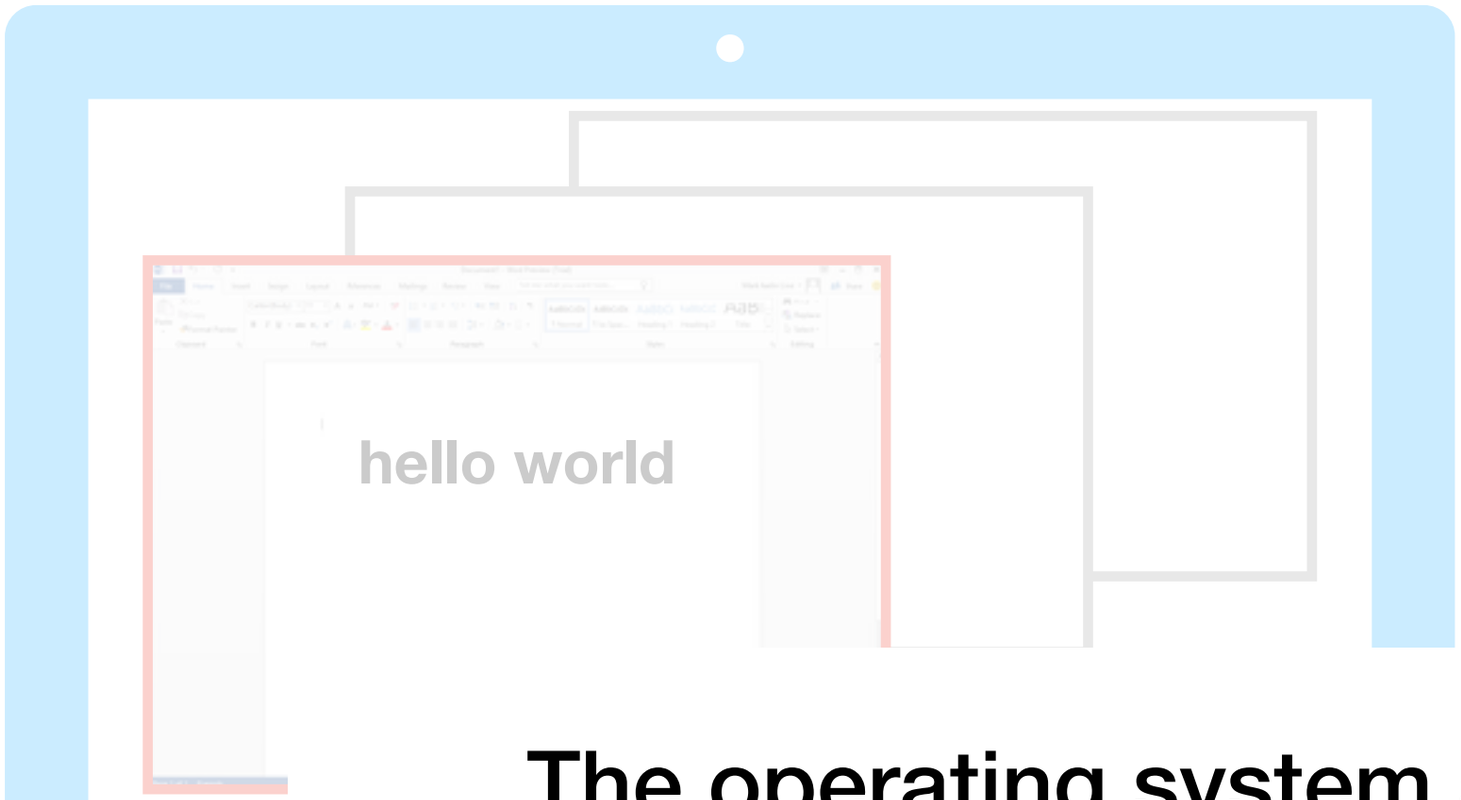
Write!
(output)

1 TB





The operating system acts as an **I/O middleman** between processes and hardware (NICs, storage devices, etc)



Operating System

The operating system acts as an **I/O middleman** between processes and hardware (NICs, storage devices, etc)

Today's Topics

Program Input/Output

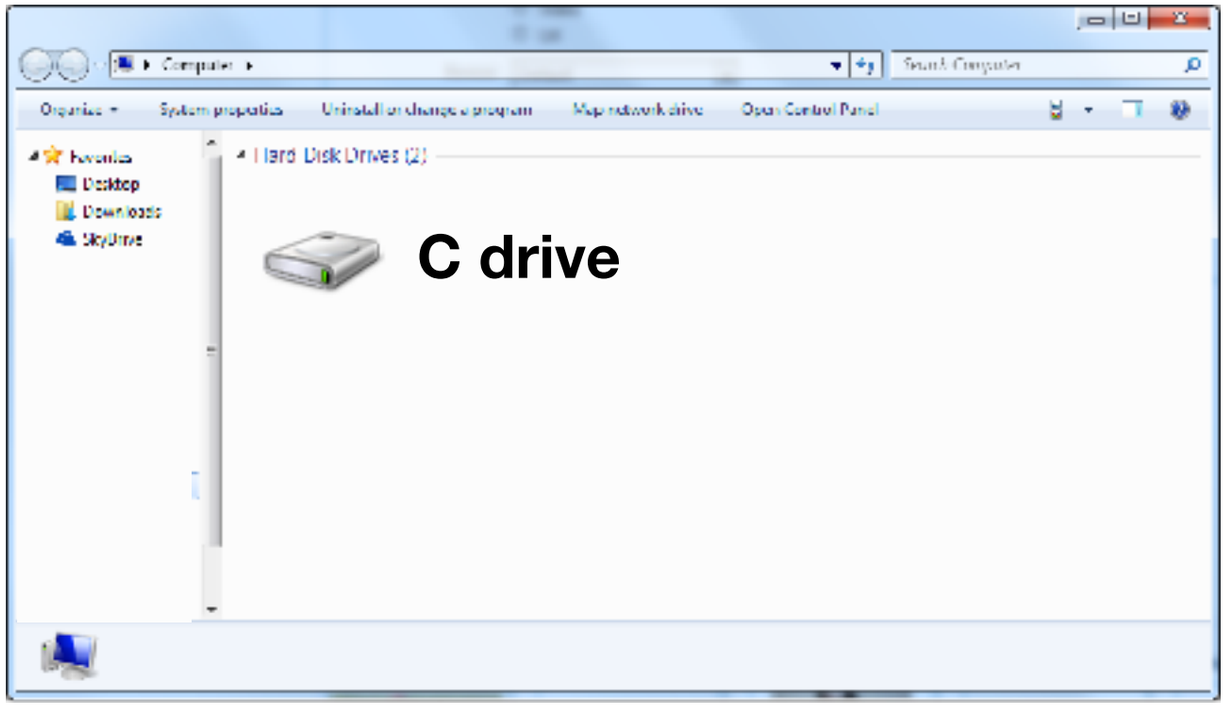
File Systems

- Storage Devices in Windows
- File I/O
- Organizing with Folders/Directories
- Storage Devices in Mac+Linux

Terminal Emulators and Shells

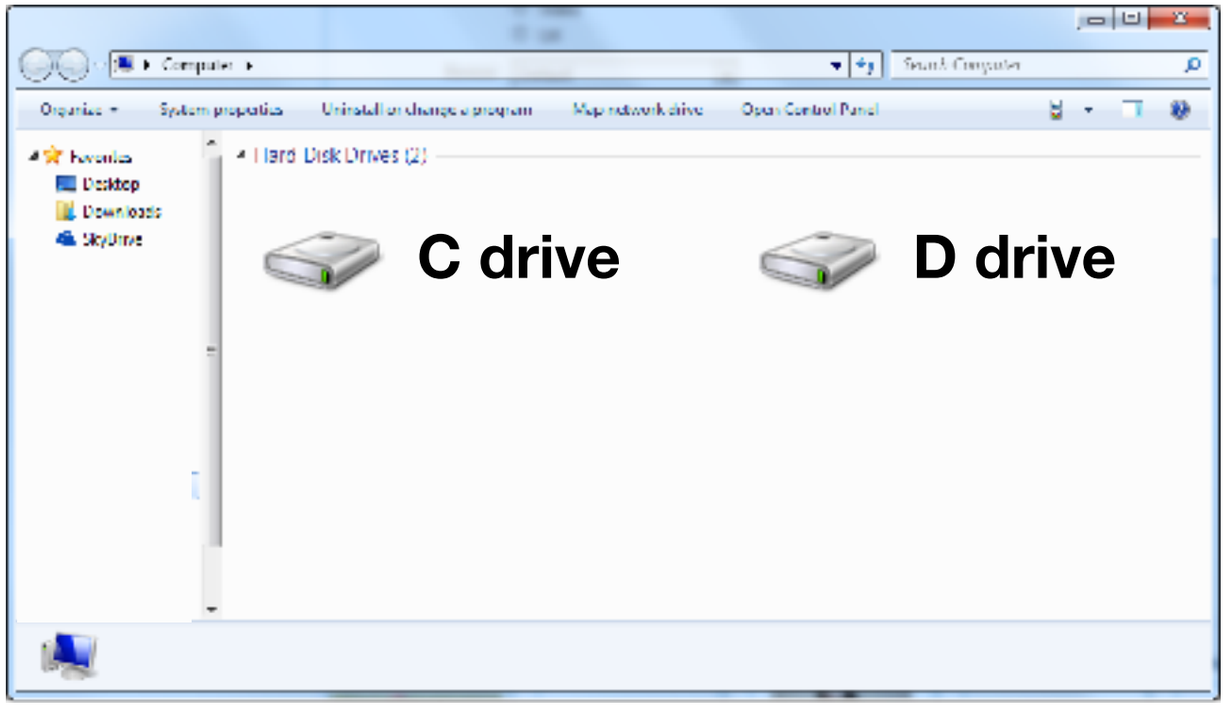
Tutorial

Windows Storage Drives



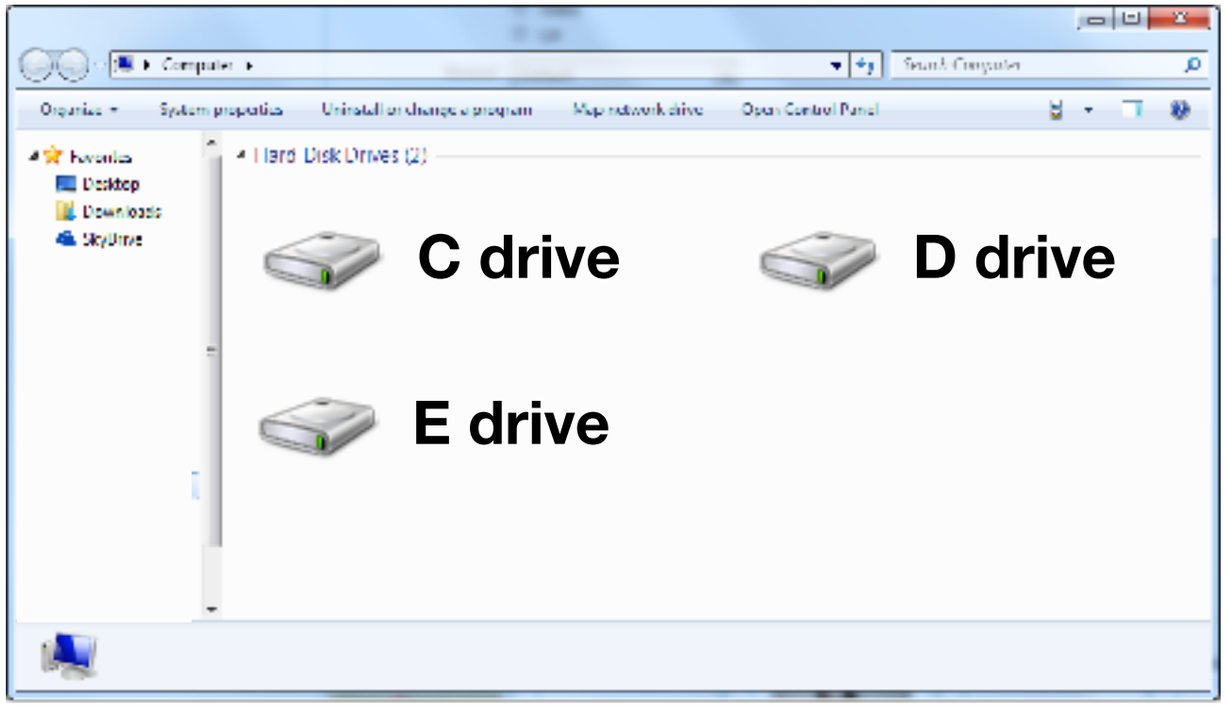
Each added drive is given its own drive letter

Windows Storage Drives



Each added drive is given its own drive letter

Windows Storage Drives



Each added drive is given its own drive letter

Today's Topics

Program Input/Output

File Systems

- Storage Devices in Windows
- File I/O
- Organizing with Folders/Directories
- Storage Devices in Mac+Linux

Terminal Emulators and Shells

Tutorial

File I/O

Each file has a name, called a “path name”

c:\README.txt

c:\hw.docx

d:\page.html

e:\main.py

File I/O

Each file has a name, called a “path name”

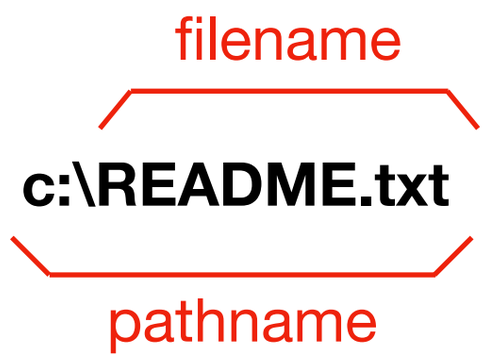
c:\hw.docx

d:\page.html

e:\main.py

File I/O

Each file has a name, called a “path name”



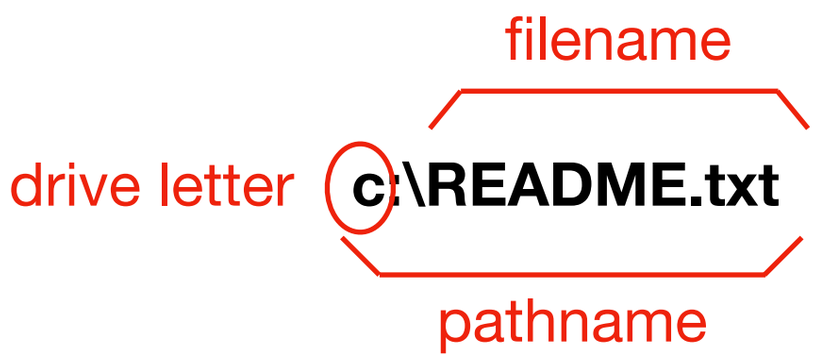
c:\hw.docx

d:\page.html

e:\main.py

File I/O

Each file has a name, called a “path name”



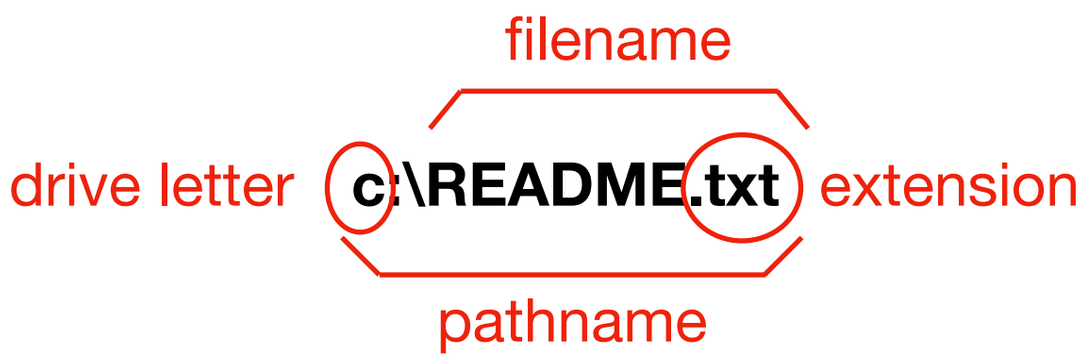
c:\hw.docx

d:\page.html

e:\main.py

File I/O

Each file has a name, called a “path name”



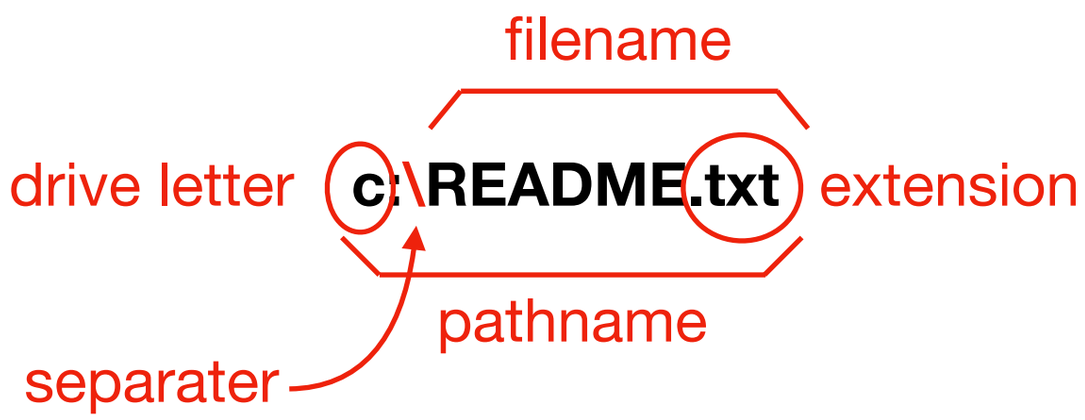
c:\hw.docx

d:\page.html

e:\main.py

File I/O

Each file has a name, called a “path name”



c:\hw.docx

d:\page.html

e:\main.py

File I/O

Files are sources of input and destinations for output for processes.

Files are managed by a part of the operating system called the “file system”

File I/O

Files are sources of input and destinations for output for processes.

Files are managed by a part of the operating system called the “file system”

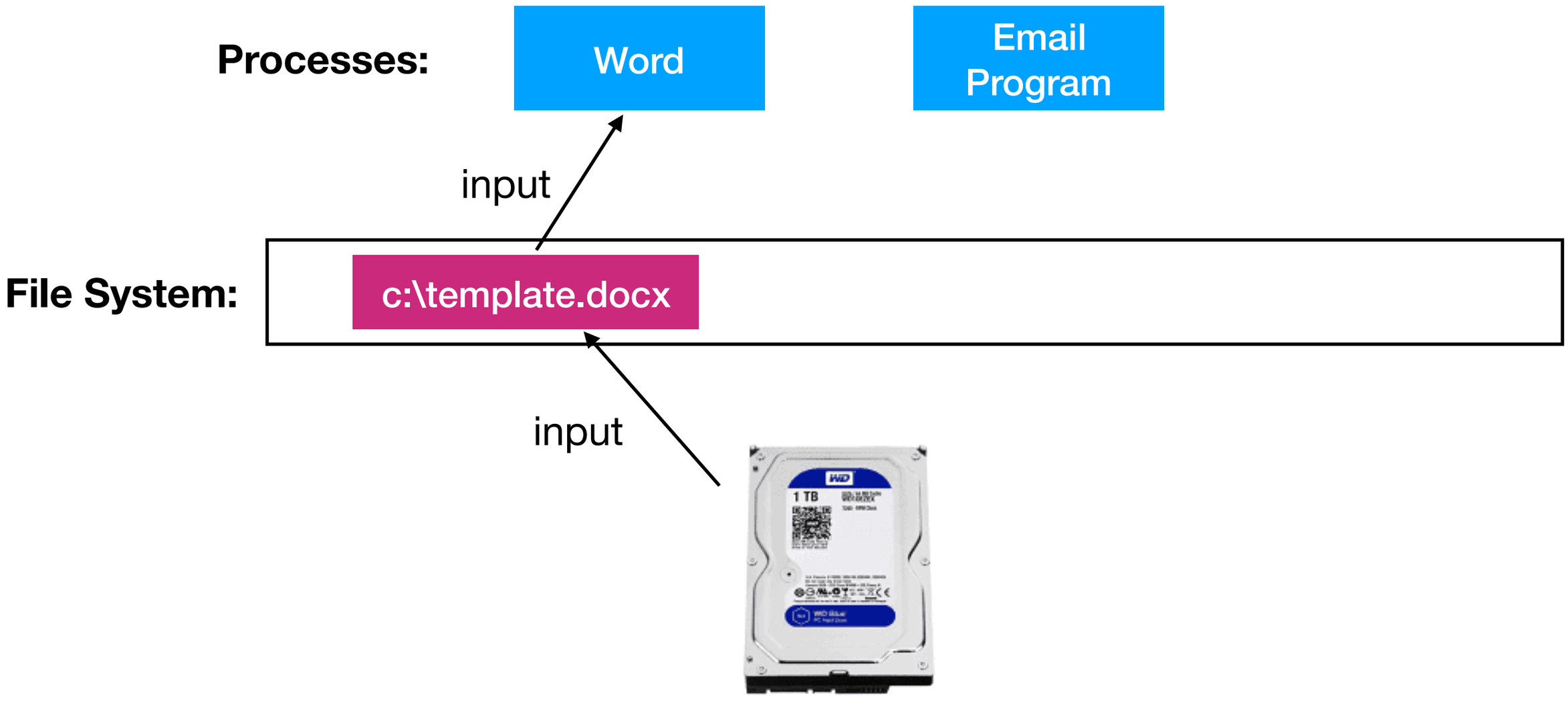
Processes: Word | Email Program

File System: c:\template.docx

File I/O

Files are sources of input and destinations for output for processes.

Files are managed by a part of the operating system called the “file system”



File I/O

Files are sources of input and destinations for output for processes.

Files are managed by a part of the operating system called the “file system”

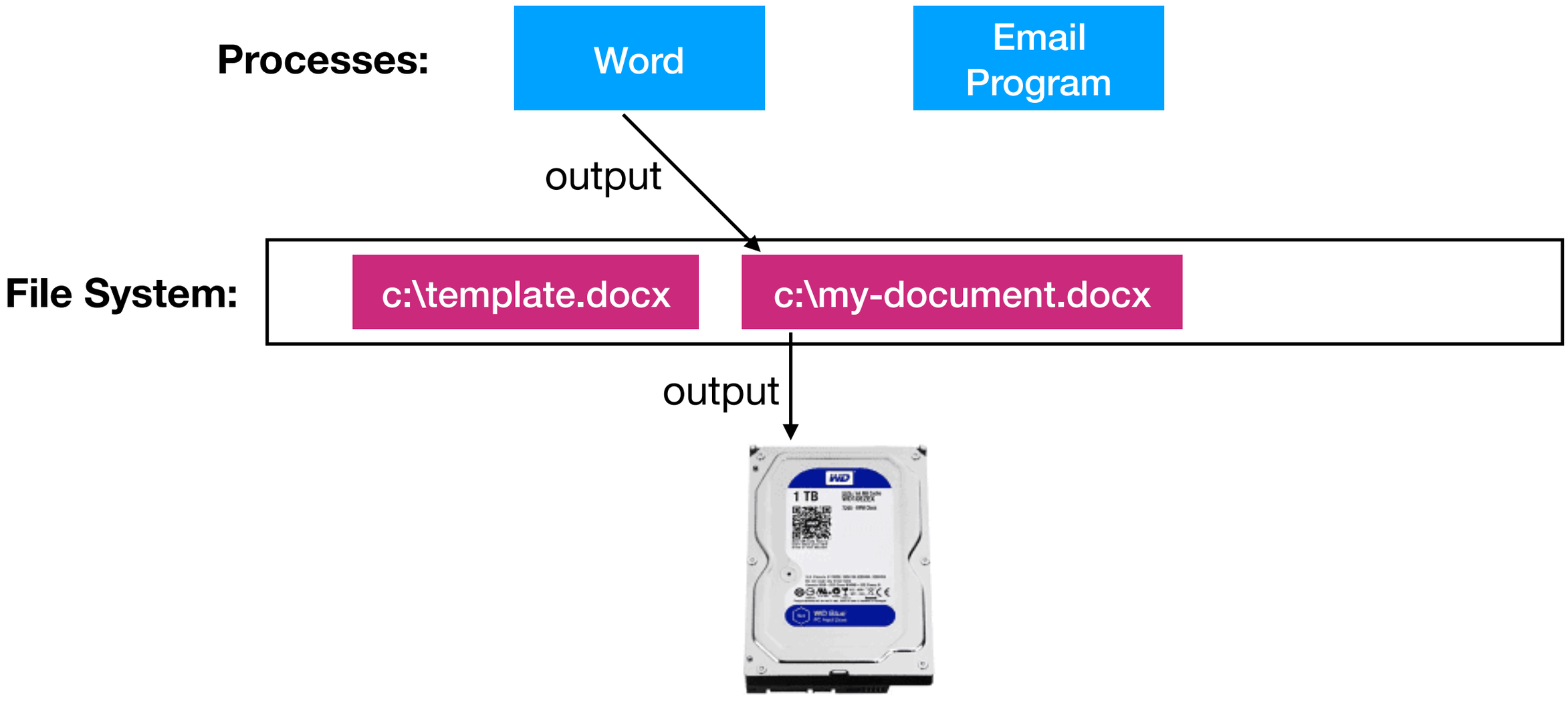
Processes: Word | Email Program

File System: c:\template.docx

File I/O

Files are sources of input and destinations for output for processes.

Files are managed by a part of the operating system called the “file system”



File I/O

Files are sources of input and destinations for output for processes.

Files are managed by a part of the operating system called the “file system”

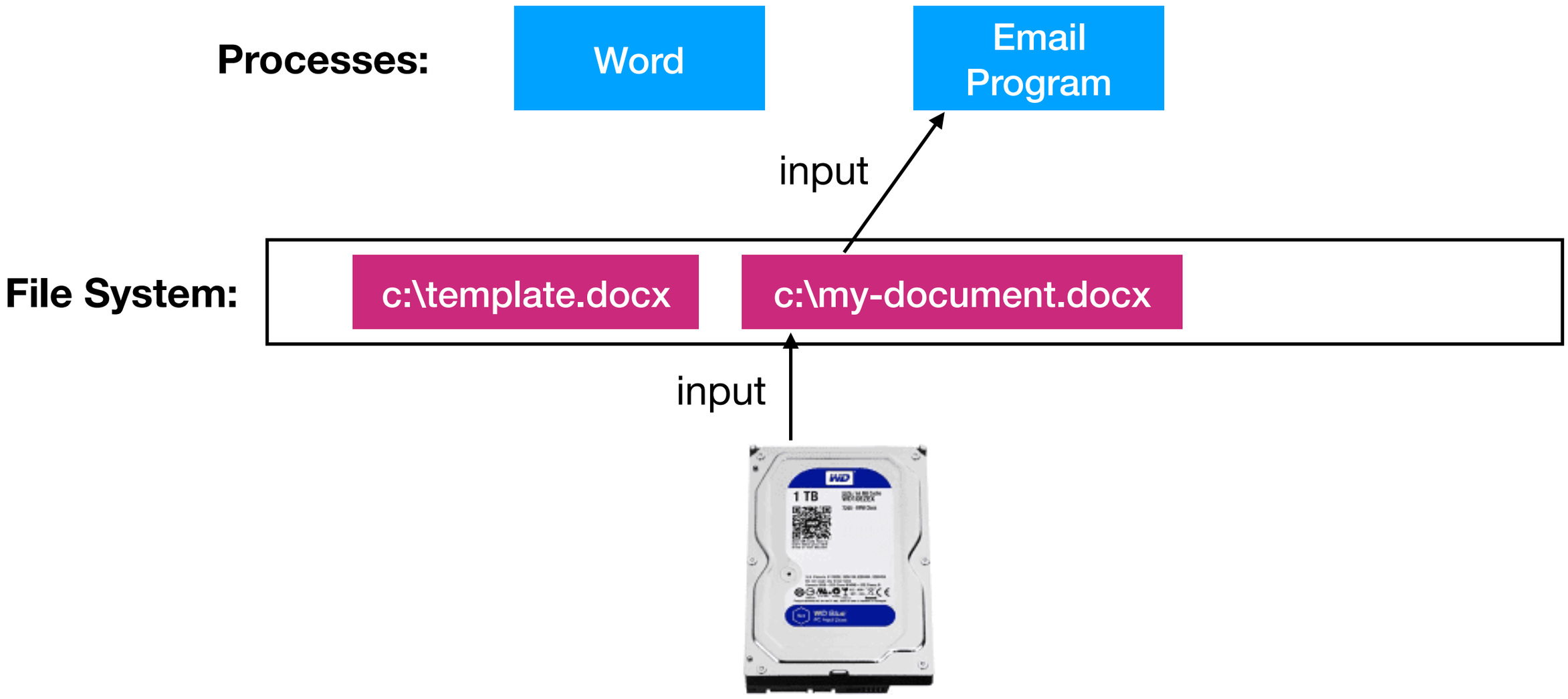
Processes: Word, Email Program

File System: c:\template.docx, c:\my-document.docx

File I/O

Files are sources of input and destinations for output for processes.

Files are managed by a part of the operating system called the “file system”



Today's Topics

Program Input/Output

File Systems

- Storage Devices in Windows
- File I/O
- Organizing with Folders/Directories
- Storage Devices in Mac+Linux

Terminal Emulators and Shells

Tutorial

Directories

Directories are used to organize files

- Also called “folders”
- A directory also has pathname
- Each directory may contain other directories and files

Example paths:

- c:\my-directory\file1.docx
- c:\my-directory\file2.docx
- c:\my-directory\file3.docx
- c:\directory1\directory2\file1.docx
- c:\same-dir\same-dir\readme.txt

Relative Paths

Where is the Computer Science building?

- Answer 1: 1210 W Dayton St, Madison, WI 53706
- Answer 2: on the other side of Johnson street

Relative Paths

Where is the Computer Science building?

- Answer 1: 1210 W Dayton St, Madison, WI 53706
- Answer 2: on the other side of Johnson street

When is Answer 2 appropriate?

Relative Paths

Where is the Computer Science building?

- Answer 1: 1210 W Dayton St, Madison, WI 53706
- Answer 2: on the other side of Johnson street

When is Answer 2 appropriate?

- When you're in the psychology building
- It may be more convenient

Relative Paths

Where is the Computer Science building?

- Answer 1: 1210 W Dayton St, Madison, WI 53706
- Answer 2: on the other side of Johnson street

When is Answer 2 appropriate?

- When you're in the psychology building
- It may be more convenient

Pathnames are absolute (answer 1) or relative (answer 2)

- Absolute paths: always possible
- Relative paths: if current location is known

Relative Paths

Where is the Computer Science building?

- Answer 1: 1210 W Dayton St, Madison, WI 53706
- Answer 2: on the other side of Johnson street

When is Answer 2 appropriate?

- When you're in the psychology building
- It may be more convenient

Pathnames are absolute (answer 1) or relative (answer 2)

- Absolute paths: always possible
- Relative paths: if current location is known
- Current location/directory is called "working directory" or "current working directory"

Absolute vs. Relative

Absolute Path	Working Directory	Relative Path
c:\test.txt	c:\	test.txt
c:\x\y\z\my.docx	c:\x\y\z	
c:\x\y\z\my.docx	c:\x\y	
c:\x\y\z	c:\x	

Absolute vs. Relative

Absolute Path	Working Directory	Relative Path
c:\test.txt	c:\	test.txt
c:\x\y\z\my.docx	c:\x\y\z	my.docx
c:\x\y\z\my.docx	c:\x\y	
c:\x\y\z	c:\x	

Absolute vs. Relative

Absolute Path	Working Directory	Relative Path
c:\test.txt	c:\	test.txt
c:\x\y\z\my.docx	c:\x\y\z	my.docx
c:\x\y\z\my.docx	c:\x\y	z\my.docx
c:\x\y\z	c:\x	

Absolute vs. Relative

Absolute Path	Working Directory	Relative Path
c:\test.txt	c:\	test.txt
c:\x\y\z\my.docx	c:\x\y\z	my.docx
c:\x\y\z\my.docx	c:\x\y	z\my.docx
c:\x\y\z	c:\x	y\z

Absolute vs. Relative

Absolute Path	Working Directory	Relative Path
c:\test.txt	c:\	test.txt
c:\x\y\z\my.docx	c:\x\y\z	my.docx
c:\x\y\z\my.docx	c:\x\y	z\my.docx
c:\x\y\z	c:\x	y\z

Two special directory names

- “..” means up a directory
- “.” means current directory

Absolute vs. Relative

Absolute Path	Working Directory	Relative Path
c:\test.txt	c:\	test.txt
c:\x\y\z\my.docx	c:\x\y\z	my.docx
c:\x\y\z\my.docx	c:\x\y	z\my.docx
c:\x\y\z	c:\x	y\z
c:\test.txt	c:\	.\test.txt
c:\x\y\z	c:\x	
c:\x	c:\x\y\z	

Two special directory names

- “..” means up a directory
- “.” means current directory

Absolute vs. Relative

Absolute Path	Working Directory	Relative Path
c:\test.txt	c:\	test.txt
c:\x\y\z\my.docx	c:\x\y\z	my.docx
c:\x\y\z\my.docx	c:\x\y	z\my.docx
c:\x\y\z	c:\x	y\z
c:\test.txt	c:\	.\test.txt
c:\x\y\z	c:\x	.\.\test.txt
c:\x	c:\x\y\z	

Two special directory names

- “..” means up a directory
- “.” means current directory

Absolute vs. Relative

Absolute Path	Working Directory	Relative Path
c:\test.txt	c:\	test.txt
c:\x\y\z\my.docx	c:\x\y\z	my.docx
c:\x\y\z\my.docx	c:\x\y	z\my.docx
c:\x\y\z	c:\x	y\z
c:\test.txt	c:\	.\test.txt
c:\x\y\z	c:\x	.\y\z
c:\x	c:\x\y\z	

Two special directory names

- “..” means up a directory
- “.” means current directory

Absolute vs. Relative

Absolute Path	Working Directory	Relative Path
c:\test.txt	c:\	test.txt
c:\x\y\z\my.docx	c:\x\y\z	my.docx
c:\x\y\z\my.docx	c:\x\y	z\my.docx
c:\x\y\z	c:\x	y\z
c:\test.txt	c:\	.\test.txt
c:\test.txt	c:\	.\.\test.txt
c:\x\y\z	c:\x	.\y\z
c:\x	c:\x\y\z	..\..

Two special directory names

- “..” means up a directory
- “.” means current directory

Absolute vs. Relative

Absolute Path	Working Directory	Relative Path
c:\test.txt	c:\	test.txt
c:\x\y\z\my.docx	c:\x\y\z	my.docx
c:\x\y\z\my.docx	c:\x\y	z\my.docx
c:\x\y\z	c:\x	y\z
c:\test.txt	c:\	.\test.txt
c:\test.txt	c:\	.\.\test.txt
c:\x\y\z	c:\x	.\y\z
c:\x	c:\x\y\z	..\..
c:\B\file.txt	c:\A	

Two special directory names

- “..” means up a directory
- “.” means current directory

Absolute vs. Relative

Absolute Path	Working Directory	Relative Path
c:\test.txt	c:\	test.txt
c:\x\y\z\my.docx	c:\x\y\z	my.docx
c:\x\y\z\my.docx	c:\x\y	z\my.docx
c:\x\y\z	c:\x	y\z
c:\test.txt	c:\	.\test.txt
c:\test.txt	c:\	.\.\test.txt
c:\x\y\z	c:\x	.\y\z
c:\x	c:\x\y\z	..\..
c:\B\file.txt	c:\A	..\B\file.txt

Two special directory names

- “..” means up a directory
- “.” means current directory

Absolute vs. Relative

Absolute Path	Working Directory	Relative Path
c:\test.txt	c:\	test.txt
c:\x\y\z\my.docx	c:\x\y\z	my.docx
c:\x\y\z\my.docx	c:\x\y	z\my.docx
c:\x\y\z	c:\x	y\z
c:\test.txt	c:\	.\test.txt
c:\x\y\z	c:\x	.\y\z
c:\x	c:\x\y\z	..\..
c:\B\file.txt	c:\A	..\B\file.txt

Two special directory names

- “..” means up a directory
- “.” means current directory

more examples in tutorial later…

Today's Topics

Program Input/Output

File Systems

- Storage Devices in Windows
- File I/O
- Organizing with Folders/Directories
- Storage Devices in Mac+Linux

Terminal Emulators and Shells

Tutorials

Multiple Drives in Linux

Windows

- Generally, every absolute pathname starts with “c:\” or “d:\” or similar
- Name indicates which drive stores the file

UNIX

- Every absolute pathname starts with “/”
- For example, /home/tyler/my-file.docx (note forward slash)
- Name does not indicate on which drive a file lives

Multiple Drives in Linux

Windows

- Generally, every absolute pathname starts with “c:\” or “d:\” or similar
- Name indicates which drive stores the file

UNIX

- Every absolute pathname starts with “/”
- For example, /home/tyler/my-file.docx (note forward slash)
- Name does not indicate on which drive a file lives

How can we use multiple drives if every file paths starts the same, with “/” ???

Multiple Drives in Linux

Windows

- Generally, every absolute pathname starts with “c:\” or “d:\” or similar
- Name indicates which drive stores the file

UNIX

- Every absolute pathname starts with “/”
- For example, /home/tyler/my-file.docx (note forward slash)
- Name does not indicate on which drive a file lives

How can we use multiple drives if every file paths starts the same, with “/” ???

Answer: different drives feel like different directories in UNIX systems

Comparison

Windows	Mac	UNIX	Drives
c:\Users\tyler\file.txt	/Users/tyler	/home/tyler	
c:\Program Files	/usr/local/bin	/usr/local/bin	
c:\Windows\…\Logs	/var/log	/var/log	
d:\	/Volumes	/mnt/backup	
d:\aug	/Volumes/backup/aug	/mnt/backup/aug	
e:\movies	/Volumes/movies	/home/tyler/movies	

Comparison

Windows	Mac	Linux	Drives
c:\Users\tyler\file.txt	/Users/tyler	/home/tyler	
c:\Program Files	/usr/local/bin	/usr/local/bin	
c:\Windows\…\Logs	/var/log	/var/log	
d:\	/Volumes	/mnt/backup	
d:\aug	/Volumes/backup/aug	/mnt/backup/aug	
e:\movies	/Volumes/movies	/home/tyler/movies	

On Mac, extra drives often appear under /Volumes.
On Linux, extra drives often appear under /mnt (for mount).

Today's Topics

Program Input/Output

File Systems

Terminal Emulators and Shells

- Terminal history
- Shells
- Running programs from a shell

Tutorials

History: the Original Terminals

Mainframe
(powerful computer)

https://techcrunch.com/2015/08/16/ibm-teams-with-canonical-on-linux-mainframe/
https://en.wikipedia.org/wiki/Computer_terminal#Dumb_terminals

History: the Original Terminals

Mainframe
(powerful computer)

How to share it?

https://techcrunch.com/2015/08/16/ibm-teams-with-canonical-on-linux-mainframe/
https://en.wikipedia.org/wiki/Computer_terminal#Dumb_terminals

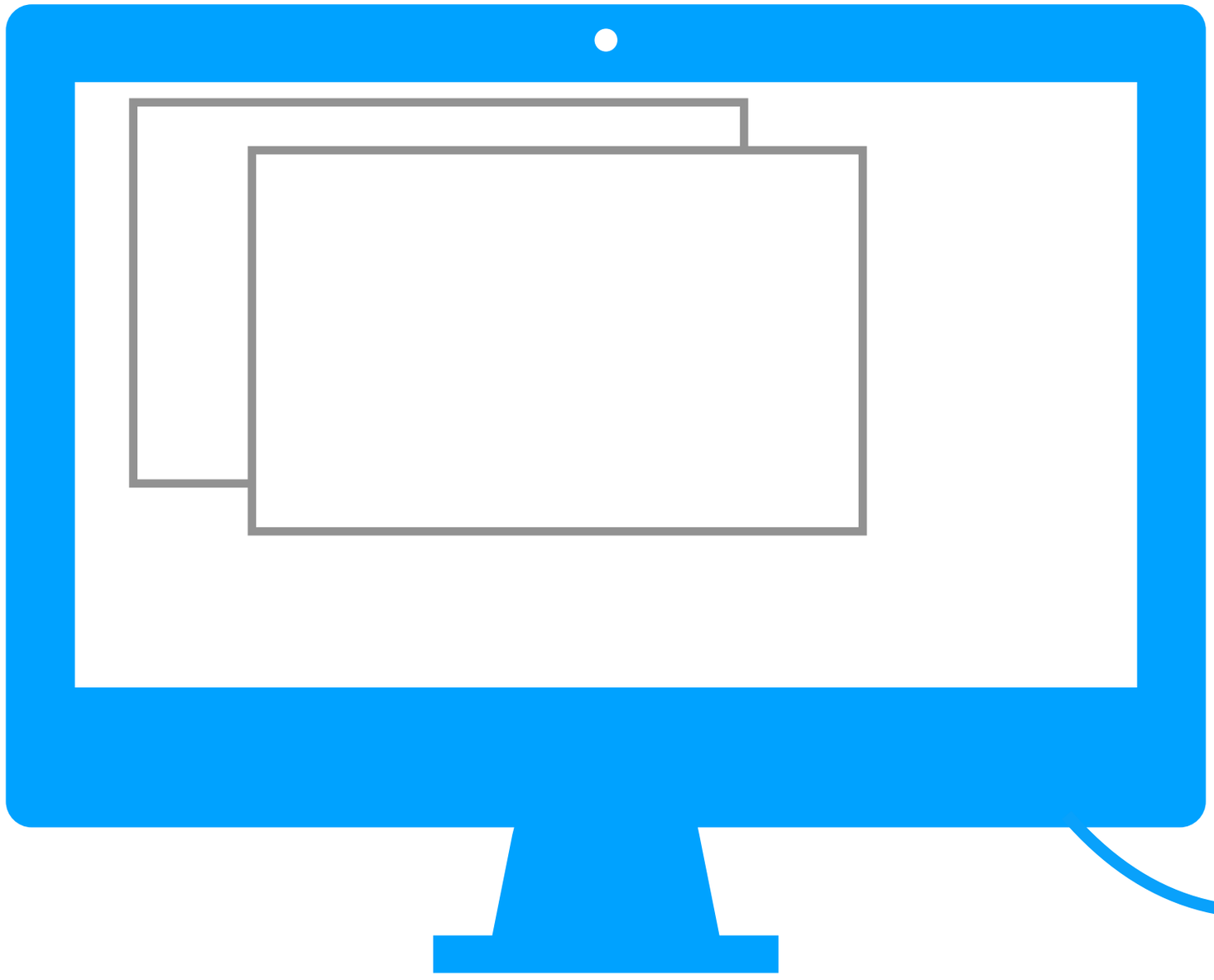
History: the Original Terminals

Mainframe
(powerful computer)

dumb terminals
(text based)

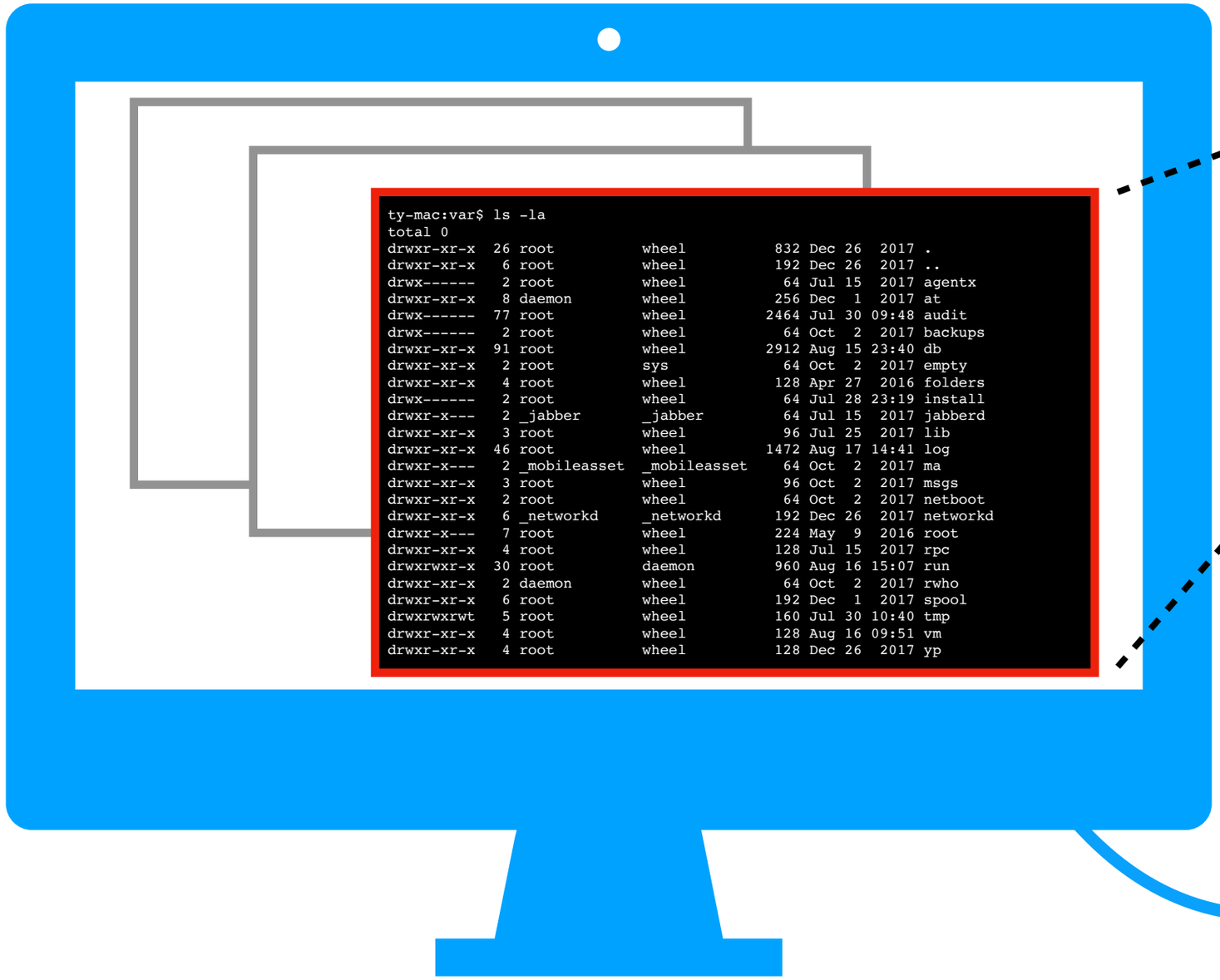
https://techcrunch.com/2015/08/16/ibm-teams-with-canonical-on-linux-mainframe/
https://en.wikipedia.org/wiki/Computer_terminal#Dumb_terminals

Terminal Emulators



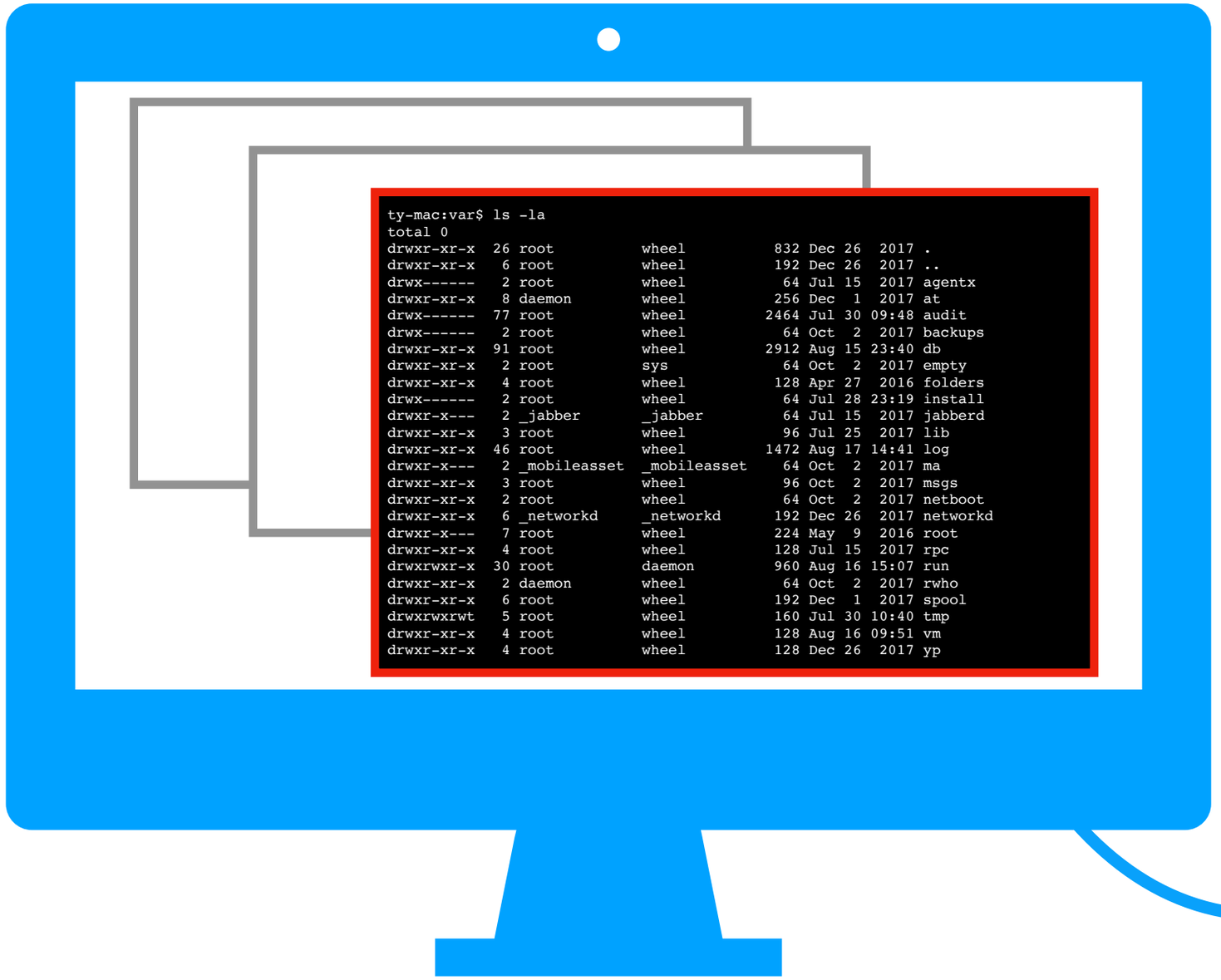
https://techcrunch.com/2015/08/16/ibm-teams-with-canonical-on-linux-mainframe/
https://en.wikipedia.org/wiki/Computer_terminal#Dumb_terminals

Terminal Emulators



https://techcrunch.com/2015/08/16/ibm-teams-with-canonical-on-linux-mainframe/
https://en.wikipedia.org/wiki/Computer_terminal#Dumb_terminals

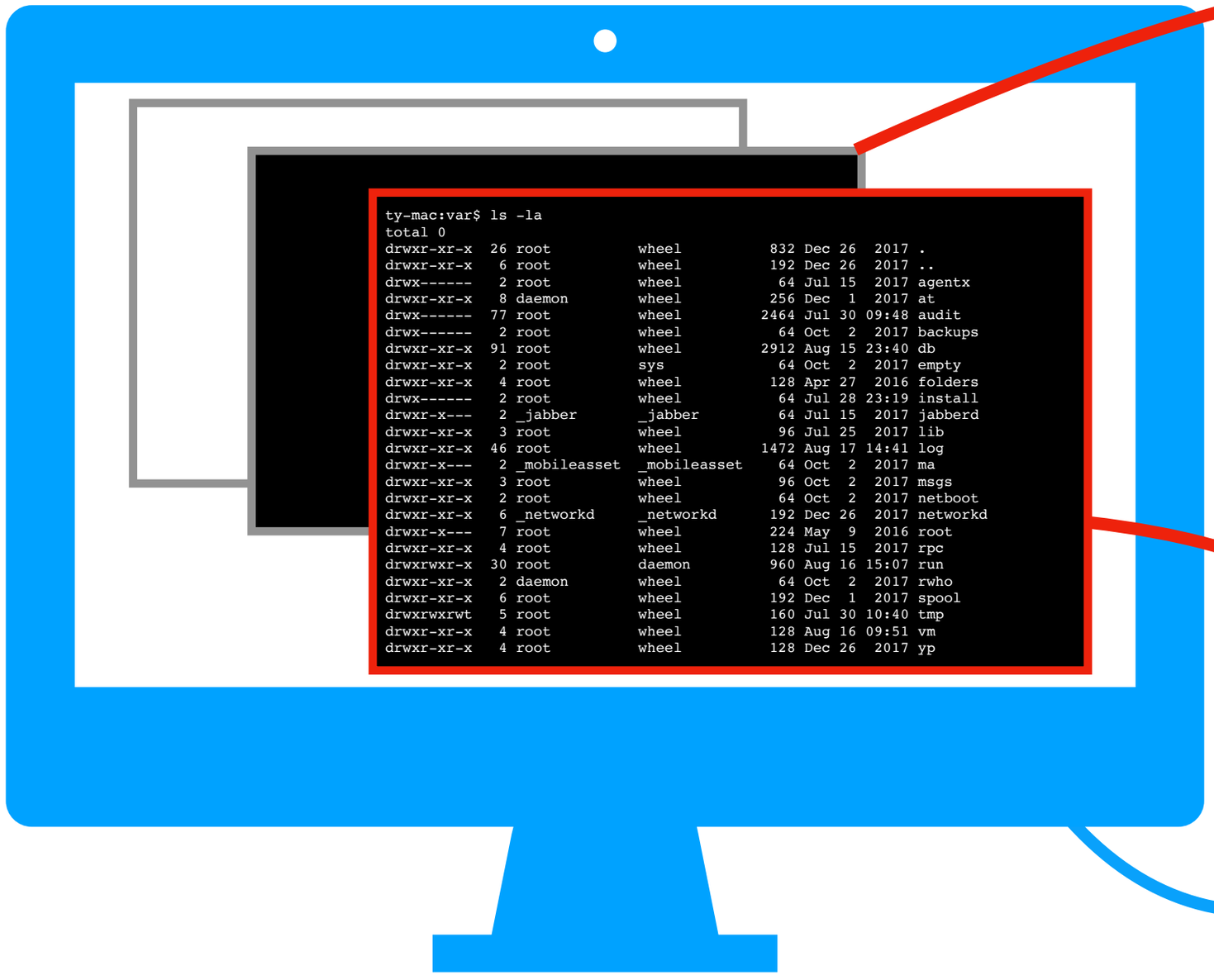
Terminal Emulators



local computer
(e.g., personal)

https://techcrunch.com/2015/08/16/ibm-teams-with-canonical-on-linux-mainframe/
https://en.wikipedia.org/wiki/Computer_terminal#Dumb_terminals

Terminal Emulators



remote computer
(e.g., CS lab)

OR

local computer
(e.g., personal)

https://techcrunch.com/2015/08/16/ibm-teams-with-canonical-on-linux-mainframe/
https://en.wikipedia.org/wiki/Computer_terminal#Dumb_terminals

Today's Topics

Program Input/Output

File Systems

Terminal Emulators and Shells

- Terminal history
- Shells
- Running programs from a shell

Tutorials

Shells

Inside a terminal, a program called a “shell” runs

- The shell lets users type commands, then carries out the appropriate actions
- Exploring files and running programs are common activities
- You will be running Python programs from a shell in a terminal!
- Different shells have minor (or major) variations

Shells

Inside a terminal, a program called a “shell” runs

- The shell lets users type commands, then carries out the appropriate actions
- Exploring files and running programs are common activities
- You will be running Python programs from a shell in a terminal!
- Different shells have minor (or major) variations

Windows Shells

- cmd
- PowerShell

UNIX Shells

- bash
- csh
- zsh
- many more

Today's Topics

Program Input/Output

File Systems

Terminal Emulators and Shells

- Terminal history
- Shells
- Running programs from a shell

Tutorials

Running Programs

Running programs is easy, just type name of the program and hit enter:

```
ty-mac:var$
```

Running Programs

Running programs is easy, just type name of the program and hit enter:

```
ty-mac:var$ ls
```

Running Programs

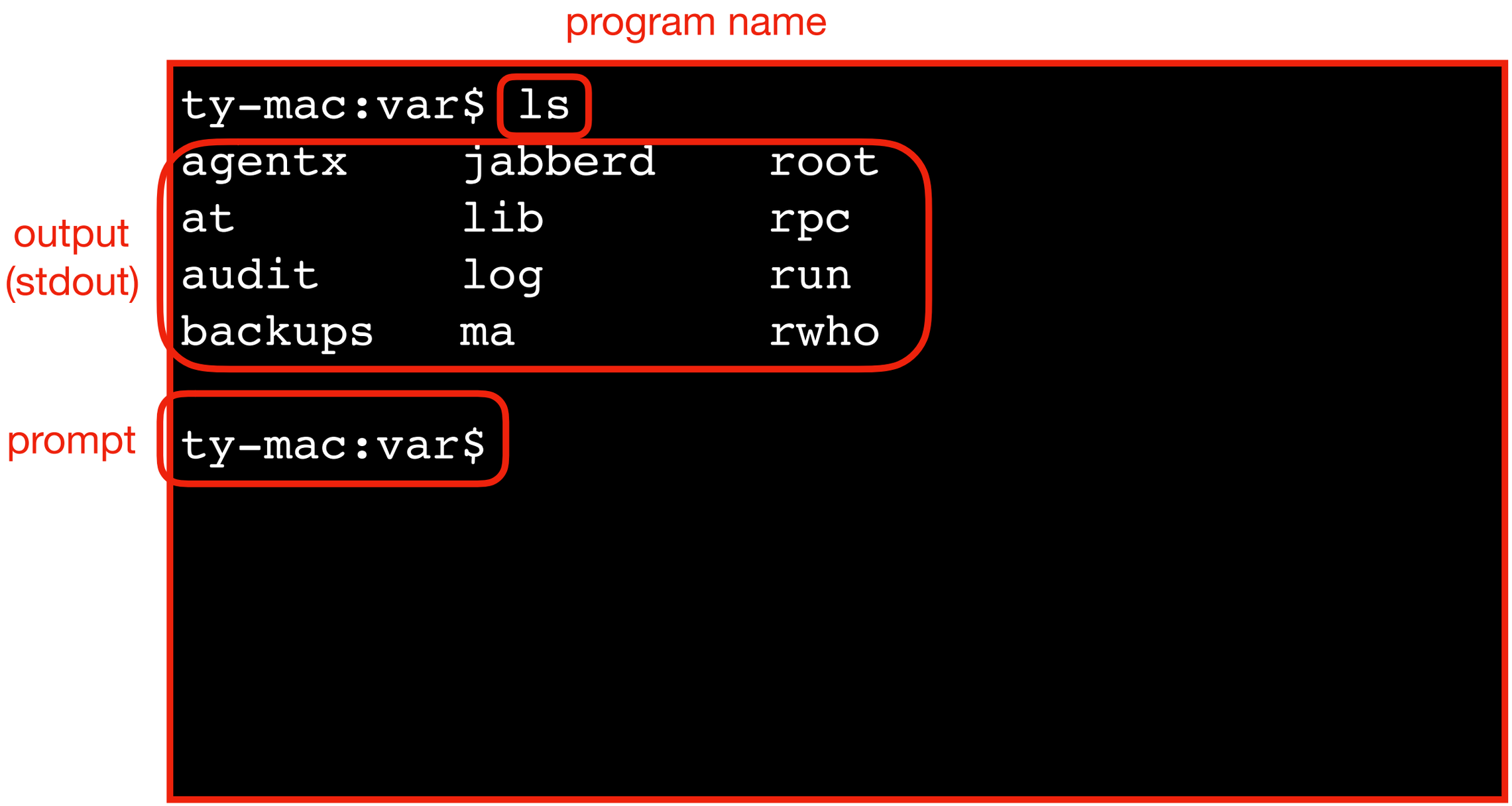
Running programs is easy, just type name of the program and hit enter:

```
ty-mac:var$ ls
agentx      jabberd     root
at          lib         rpc
audit       log         run
backups     ma          rwho

ty-mac:var$
```

Running Programs

Running programs is easy, just type name of the program and hit enter:



Today's Topics

Program Input/Output

File Systems

Terminal Emulators and Shells

Tutorials

- PowerShell
- bash
- Cover navigation, arguments, redirects, pipes, and scripts

Conclusion

Today we covered

- File systems
- Terminals
- Shells
- Navigating in PowerShell and bash
- Redirection and piping