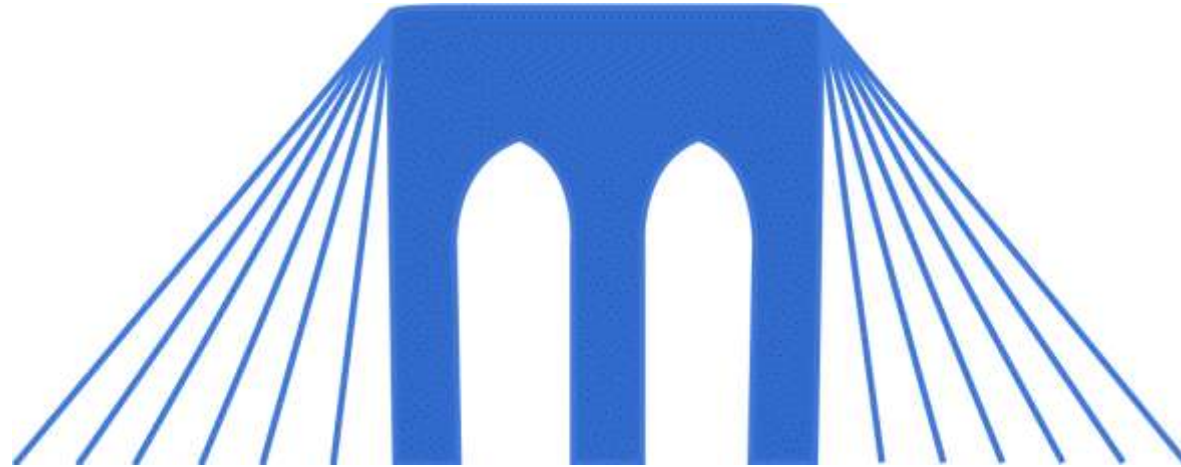


# BRIDGES TO COMPUTING



## General Information:

- This document was created for use in the "Bridges to Computing" project of Brooklyn College.
- You are invited and encouraged to use this presentation to promote computer science education in the U.S. and around the world.
- For more information about the Bridges Program, please visit our website at: <http://bridges.brooklyn.cuny.edu/>

## Disclaimers:

- All images in this presentation were created by our Bridges to Computing staff or were found online through open access media sites and are used under the Creative Commons Attribution-Share Alike 3.0 License.
- If you believe an image in this presentation is in fact copyrighted material, never intended for creative commons use, please contact us at <http://bridges.brooklyn.cuny.edu/> so that we can remove it from this presentation.
- This document may include links to sites and documents outside of the "Bridges to Computing" domain. The Bridges Program cannot be held responsible for the content of 3<sup>rd</sup> party sources and sites.

# Introduction to Operating Systems (OS)

*"Hey, hey, hey, it's the big 'Master Control Program' everybody's talking about."*

-- Kevin Flynn, (Tron 1982)



# Not every computer has an OS

- Computers are everywhere!
  - Watches
  - Phones
  - Appliances (Microwaves, Refrigerators)
  - Vehicles (All modern cars, buses, motorcycles)
  - Pacemakers
  - RFID tags
- If a computer only one set of tasks to perform and uses very simple input (a few buttons) it usually doesn't need an operating system.
- Simple computers run one program at a time, and that program talks directly to the hardware.

# What is an OS?

- A computer program (also a software program, or program) is a sequence of instructions written to perform a specified task for a computer.

An operating system is a ***special type of program*** designed to run other programs on a computer.

- The "Universal Machine" was a long sought goal of early computer scientists. Before operating systems were developed a programmer:
  - Had to know the exact hardware details of the machine they were working on.
  - Had to recompile and reload entire program to make even trivial changes.
  - Couldn't really reuse functionality (keyboard, mouse).

# The most important program

- A computer's OS is its most important program.
  - The OS is the backbone of a computer, managing both software (other programs) and hardware resources.
  - An OS is responsible for everything from the control and allocation of memory, to recognizing input from external devices and transmitting output to computer displays.
  - They also manage files on computer hard drives and control peripherals, like printers and scanners.
- An operating system is usually stored on the hard-drive (when the machine is off) and is loaded into main RAM by the BIOS on the motherboard.

# Modern Operating Systems

- The operating system of a large computer system does more than just manage hardware and program.
  - A modern OS may monitor many different programs and users, making sure everything runs smoothly, without interference, despite the fact that numerous devices and programs are used simultaneously.
  - An operating system also has a vital role to play in security. Its job includes preventing unauthorized users from accessing the computer system.

# Multi - Systems

- There are multiuser, multiprocessing, multitasking, multithreading, and real-time operating systems.
- A multiuser operating system enables multiple users to run programs simultaneously.
  - This type of operating system may be used for just a few people or hundreds of them.
  - There are some operating systems that are used to allow thousands of people to run programs at the same time.

# Multi - Systems (2)

- A multiprocessing or multiprocessor, operating system allows programs to run on more than one central processing unit (CPU) at a time.
  - This can come in very handy in some work environments, at schools, and even for some home-computing situations.
  - Of course you need to actually have multiple cpus.
- Multitasking operating systems work a little differently; they make it possible to run more than one program at a time.
- Multithreading operating systems are even more different, allowing varied parts of one program to be used simultaneously.



# Real-time Systems

- Real-time operating systems are designed to allow computers to process and respond to input instantly.
  - Usually, general-purpose operating systems, such as disk operating system (DOS), are not considered real time, as they may require seconds or minutes to respond to input.
  - Real-time operating systems are typically used when computers must react to the consistent input of information without delay.
- Examples:
  - Real-time operating systems are used in navigation, medicine and energy fields.

# Command Line Interfaces (CLI)

- A command-line interface (CLI) is a mechanism (sometimes a program) that is used to interact with a program or operating system.
- A CLI requires specific commands to be typed in by the user to perform specific tasks.
- Programs can be written to run inside a CLI. These programs (sometimes called "batch scripts") can be used to automate common tasks.
- There are many different kinds of CLI.
  - On windows there is [cmd.exe](#)
  - On MAC and Linux machines there are the "shells" sh, ksh, csh, tcsh, and bash.

# BASH and CMD.exe CLIs

```
Command Prompt - bash.exe
bash$ help
GNU bash, version 4.1.17(1)
Shell commands that are defined internally. Type 'help' to see this list.
Type 'help name' to find out more about the function 'name'.
Use 'info bash' to find out more about the shell in general.

A star (*) next to a name means that the command is disabled.

%?[DIGITS | WORD] [8]          . [filename]
:                               Some variable names and meanings
[ arg... ]                    alias [ name[=value] ... ]
bind [-lvd] [-m keymap] [-f file] break [n]
builtin [shell-builtin [arg ...]] case WORD in [PATTERN | PATTERN].
cd [dir]                       command [-pUv] [command [arg ...]]
continue [n]                   declare [-frxi] name[=value] ...
dirs [-l]                      echo [-neE] [arg ...]
enable [-n] [name ...]        eval [arg ...]
exec [ [-] file [redirection ...]] exit [n]
export [-n] [-f] [name ...] or export fc [-e ename] [-nlr] [first] [last]
for NAME [in WORDS ... ;] do COMMA function NAME { COMMANDS ; } or NAME
getopts optstring name [arg]   hash [-r] [name ...]
help [pattern ...]            history [n] [ [-awrn] [filename]]
if COMMANDS; then COMMANDS; [ elif let arg [arg ...]
local name[=value] ...        logout
popd [+n | -n]                pushd [dir | +n | -n]
pwd                            read [-r] [name ...]
readonly [-n] [-f] [name ...] or r return [n]
set [--abefhknrtuwxldHCP] [-o opti shift [n]
source filename               test [expr]
times                        trap [arg] [signal_spec]
type [-all] [-type | -path] [name typeset [-frxi] name[=value] ...
ulimit [-SHacdmstfpnuv [limit]] umask [-S] [mode]
unalias [-a] [name ...]      unset [-f] [-v] [name ...]
until COMMANDS; do COMMANDS; done wait [n]
while COMMANDS; do COMMANDS; done ( COMMANDS )
bash$
```

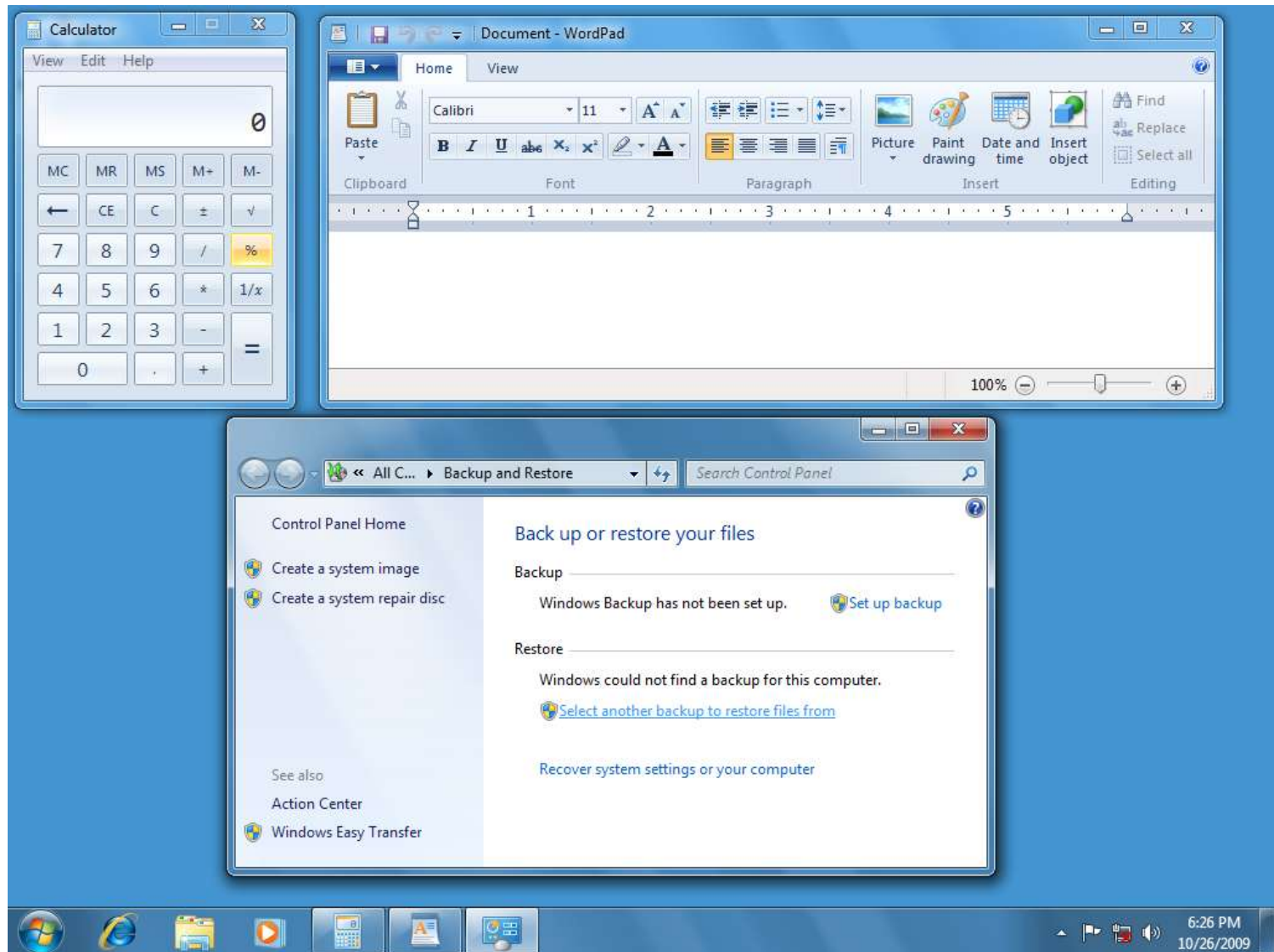
```
Command Prompt
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\matt>help
For more information on a specific command, type HELP command-name
ASSOC Displays or modifies file extension associations.
ATTRIB Displays or changes file attributes.
BREAK Sets or clears extended CTRL+C checking.
BCDEDIT Sets properties in boot database to control boot loading.
CACLS Displays or modifies access control lists (ACLs) of files
CALL Calls one batch program from another.
CD Displays the name of or changes the current directory.
CHCP Displays or sets the active code page number.
CHDIR Displays the name of or changes the current directory.
CHKDSK Checks a disk and displays a status report.
CHKNTFS Displays or modifies the checking of disk at boot time.
CLS Clears the screen.
CMD Starts a new instance of the Windows command interpreter.
COLOR Sets the default console foreground and background colors.
COMP Compares the contents of two files or sets of files.
COMPACT Displays or alters the compression of files on NTFS parti
CONVERT Converts FAT volumes to NTFS. You cannot convert the
current drive.
COPY Copies one or more files to another location.
DATE Displays or sets the date.
DEL Deletes one or more files.
DIR Displays a list of files and subdirectories in a director
DISKCOMP Compares the contents of two floppy disks.
DISKCOPY Copies the contents of one floppy disk to another.
DISKPART Displays or configures Disk Partition properties.
DOSKEY Edits command lines, recalls Windows commands, and
creates macros.
DRIVERQUERY Displays current device driver status and properties.
ECHO Displays messages, or turns command echoing on or off.
ENDLOCAL Ends localization of environment changes in a batch file.
ERASE Deletes one or more files.
EXIT Quits the CMD.EXE program (command interpreter).
FC Compares two files or sets of files, and displays the
```

# The GUI

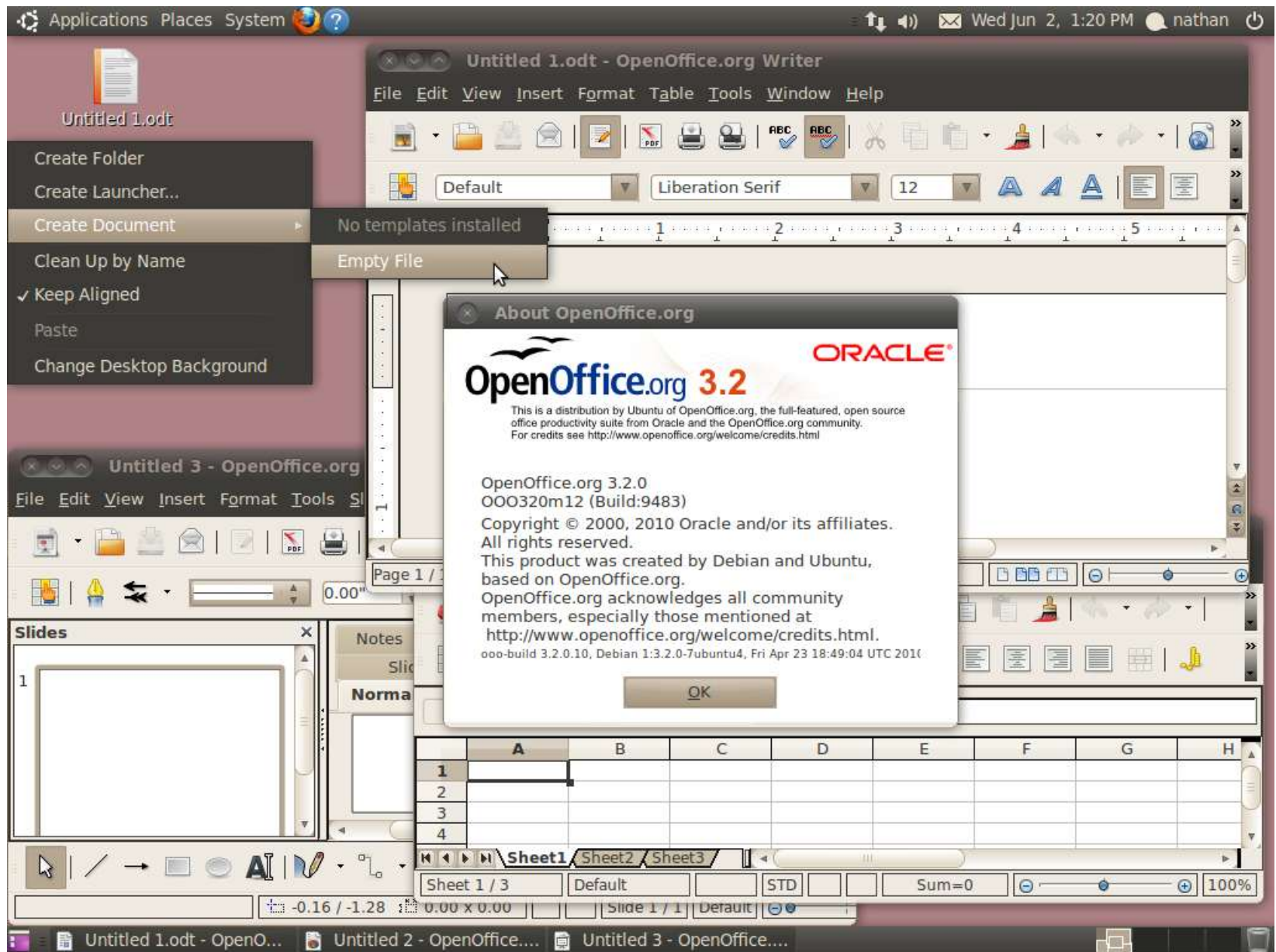
- Almost all operating systems tend include or support a graphical user interfaces (GUIs) that allow the use of devices for input. A mouse is an example of such a pointing device, as is a stylus.
- In many (non-windows) systems this GUI is a separate program, that can be stopped, started and modified. Most linux systems can run many, many different GUI's:
  - Gnome
  - KDE
  - Fluxbox
  - XFCE
  - icewm
  - Windowmaker
  - Ion

# MS Windows 7





# Ubuntu 10.04 with Gnome 2.30



# Common Operating Systems

- Commonly used operating systems for IBM-compatible personal computers include Microsoft Windows Variants, Linux Variants, and Unix Variants.
- For Macintosh computers, Mac OS X, Linux, BSD, and some Windows variants are commonly used.
- Cellphones and other portable devices usually have their own custom build Operating Systems (Apple) but other portable OS are coming online (Android).

# Application Program Interface

- A consistent application program interface (API) allows a programmer to write a program that will run on different computers, even if the hardware (memory, CPU, etc.) is different on the two machines.
- Applications continue to run when hardware upgrades and updates occur because the operating system -- not the application -- is charged with managing the hardware and the distribution of its resources.
- Today's systems can accommodate thousands of different, CPU's printers, disk drives and special peripherals in any possible combination



# Key Points

- Operating Systems manage the hardware (especially the CPU).
- Operating Systems manage the software (other programs on the machine).
- Operating Systems provide the User Interface (usually graphical, GUI) that allow human beings to interact with the computer.
- Finally, Operating Systems provide a consistent environment for programmers to work within.

# 6 primary functions

1. Processor management
2. Memory management
3. Device management
4. Storage management
5. Application interface
6. User interface



**The End**