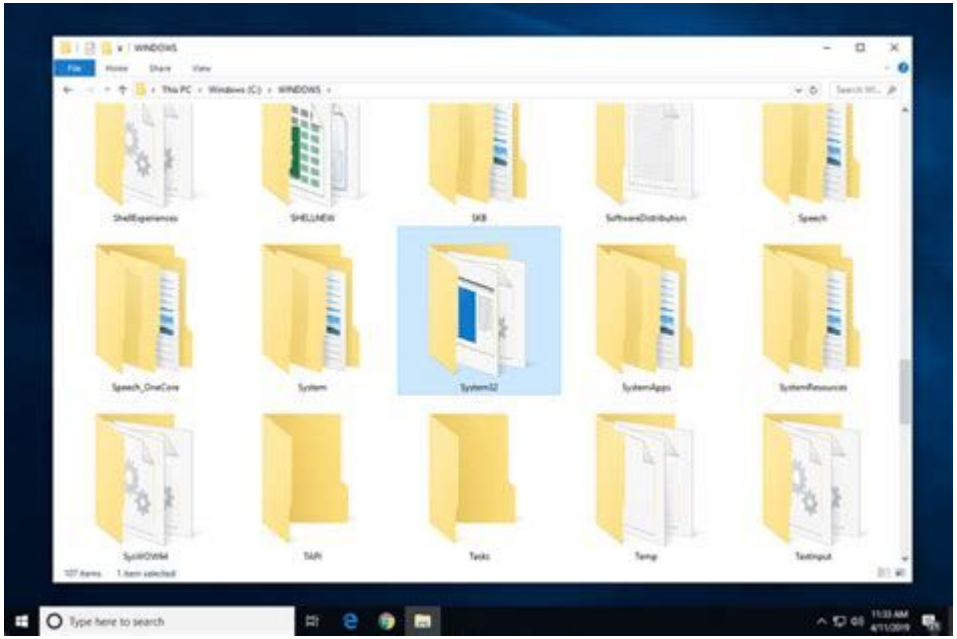
Filesystem and Paths

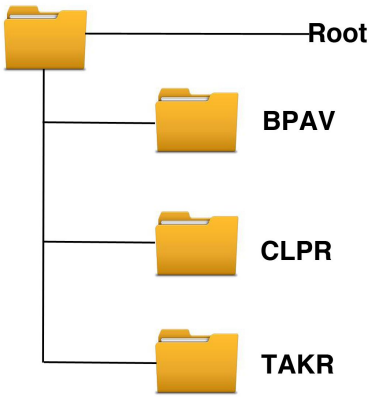
Absolute and Relative Paths

What is a filesystem?

What is a Filesystem?

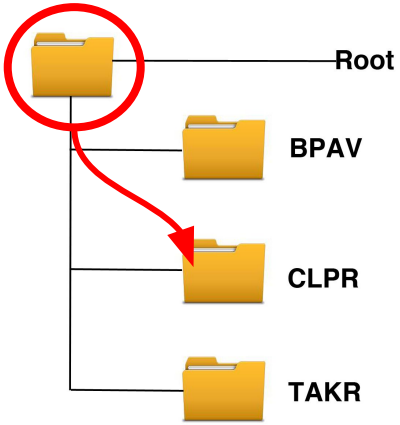
The way files and folders are organized on an operating system.





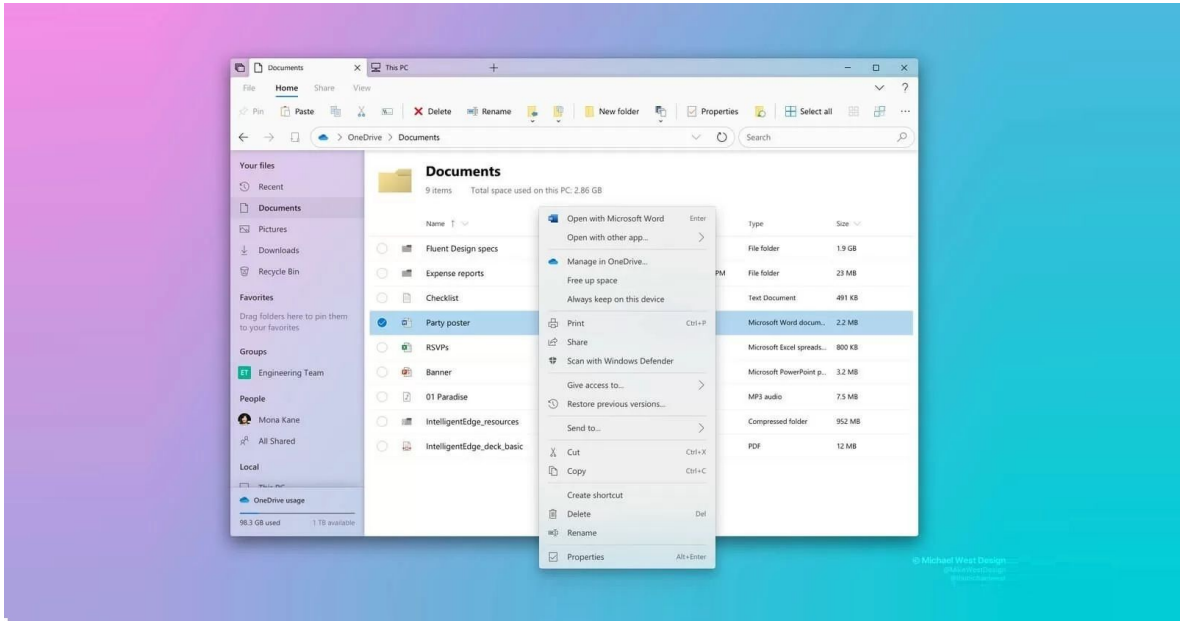
What is a Filesystem?

Many times we have folder inside folders.



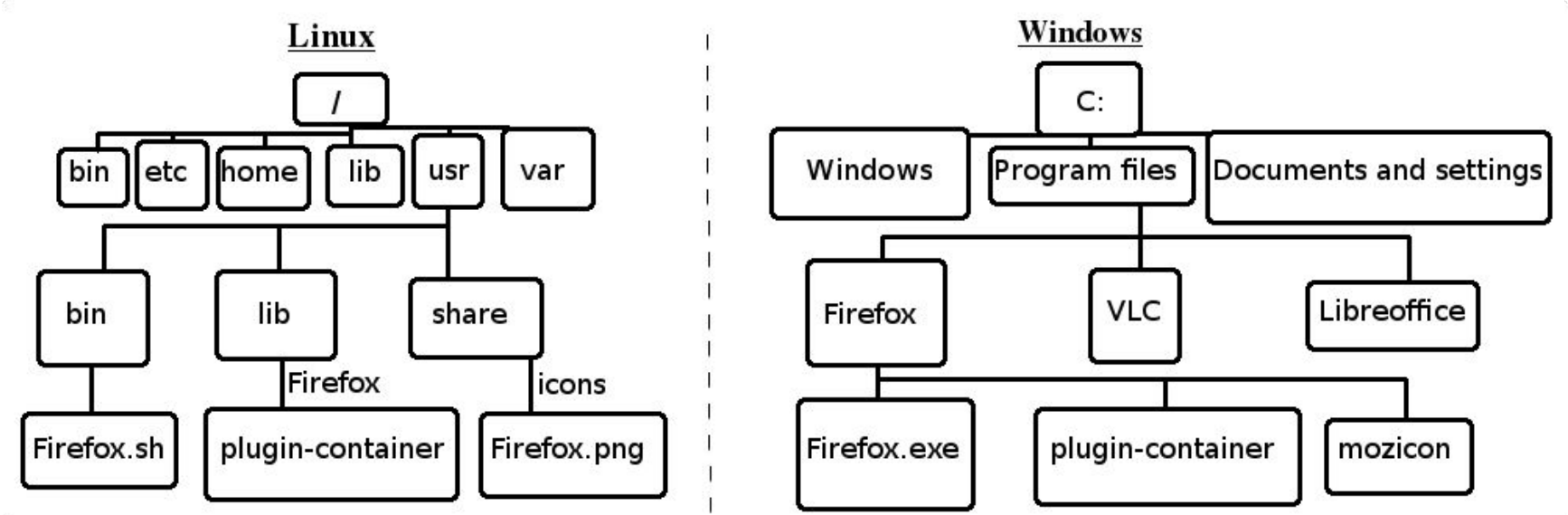
What is a Filesystem?

We usually use a file manager to move and copy files and folders around our system by dragging and dropping or cut and pasting.

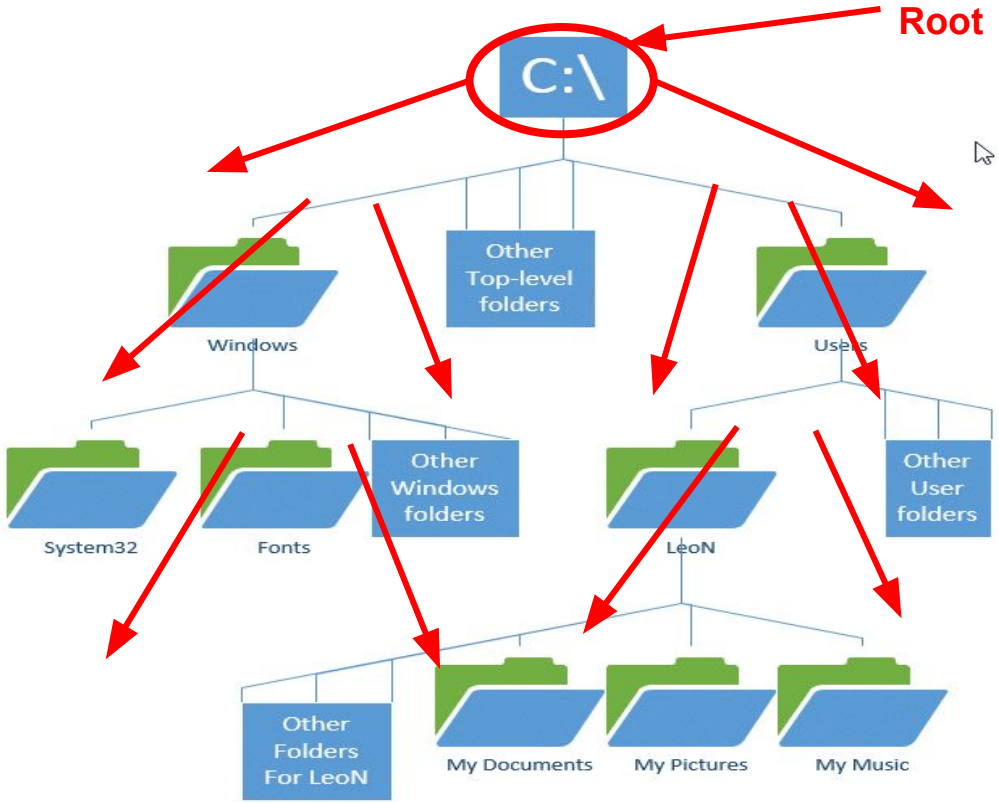


What is a Filesystem?

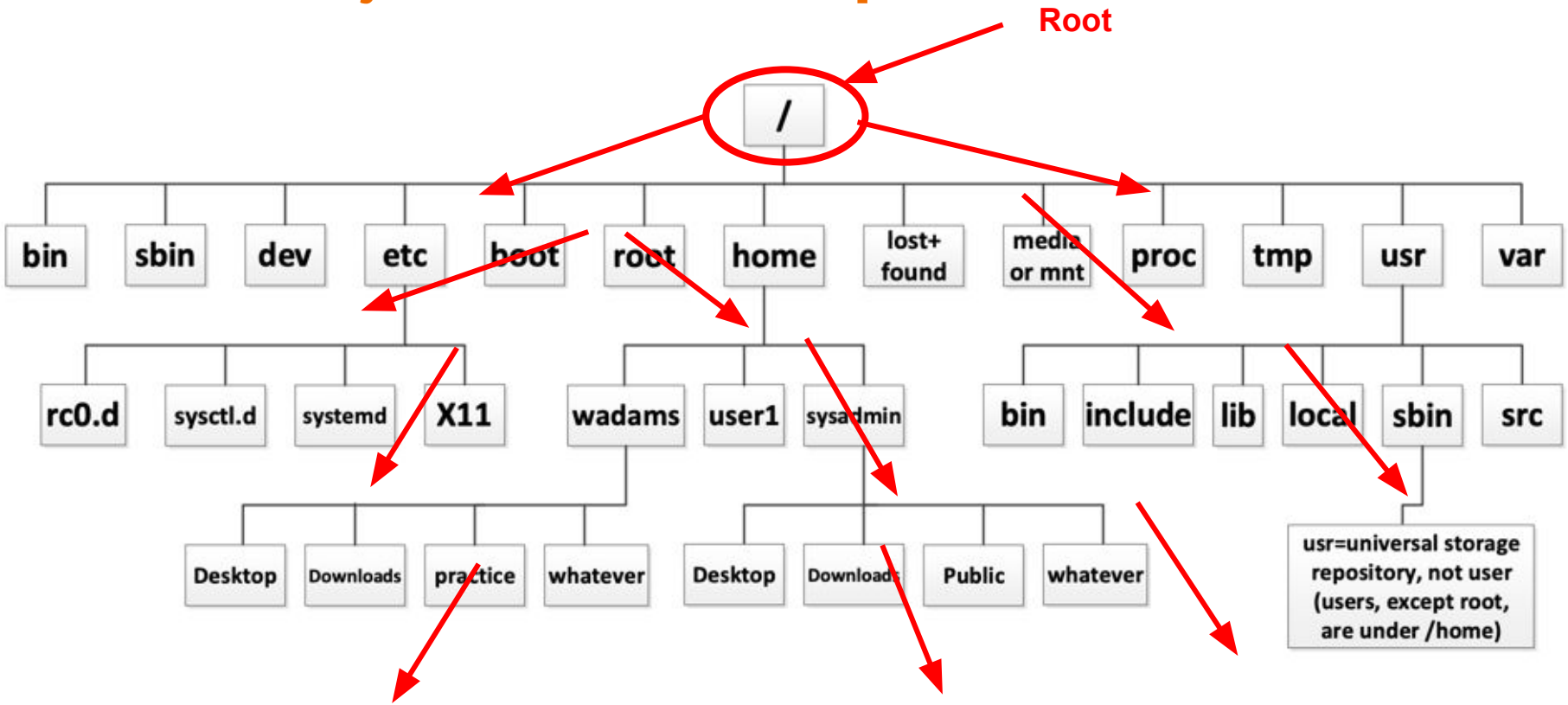
All Operating Systems organize the filesystem (files and folders) in a similar way.



Simplified Windows Filesystem

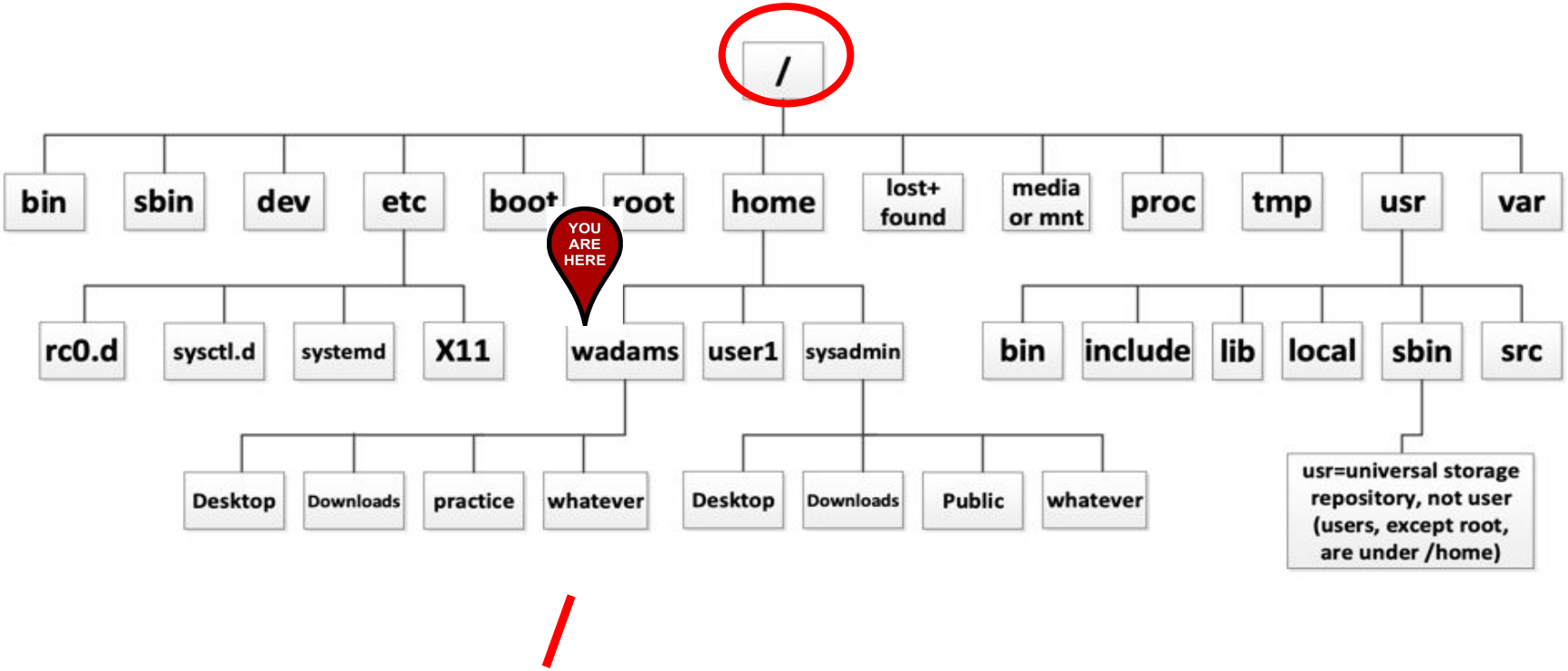


Linux Filesystem - Like an Upside Down Tree

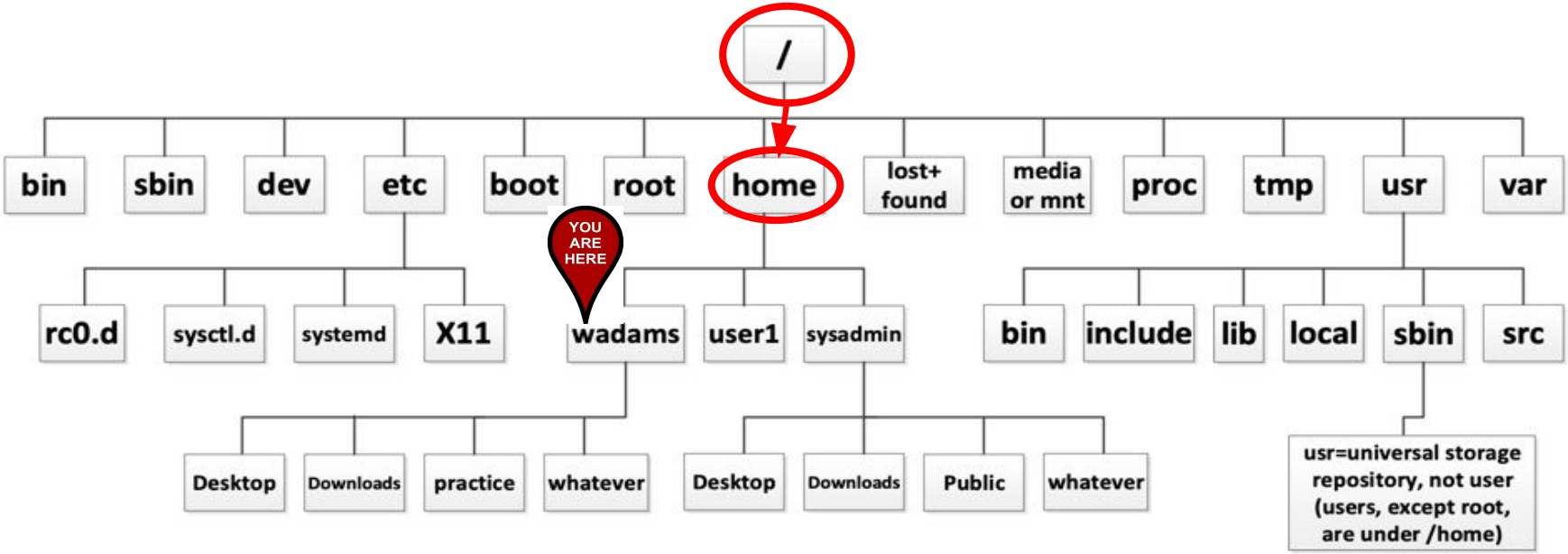


What is a path?

Path = Location

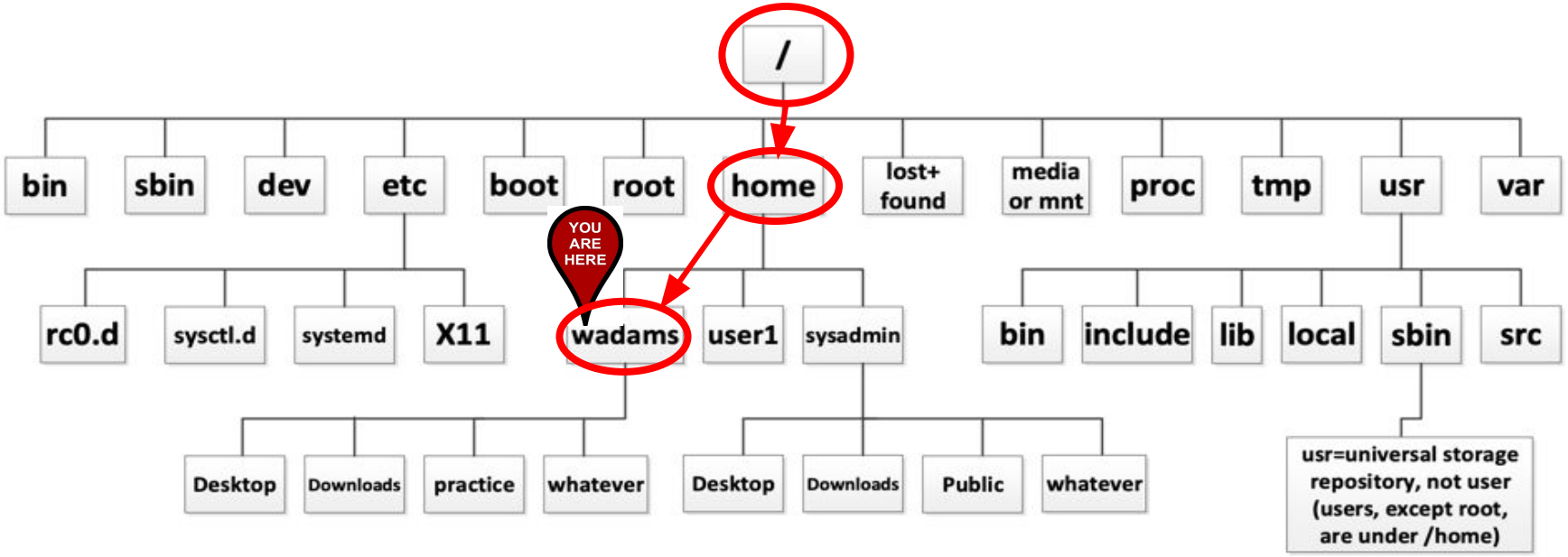


Path = Location



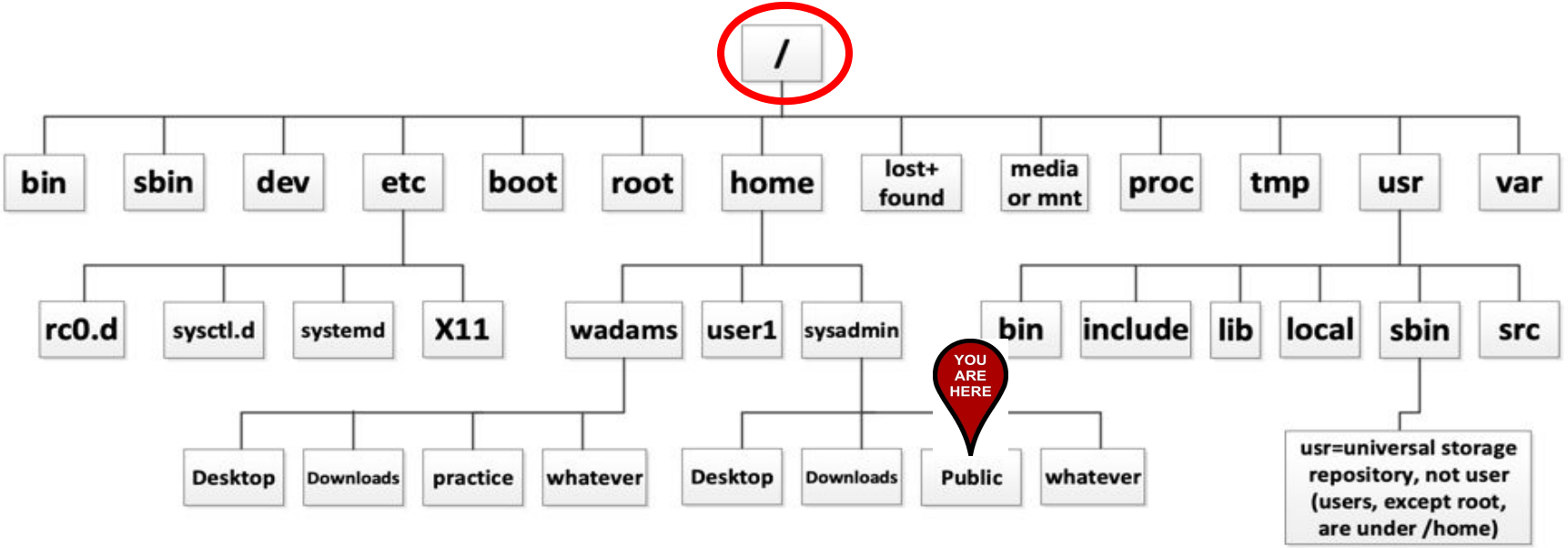
/home

Path = Location



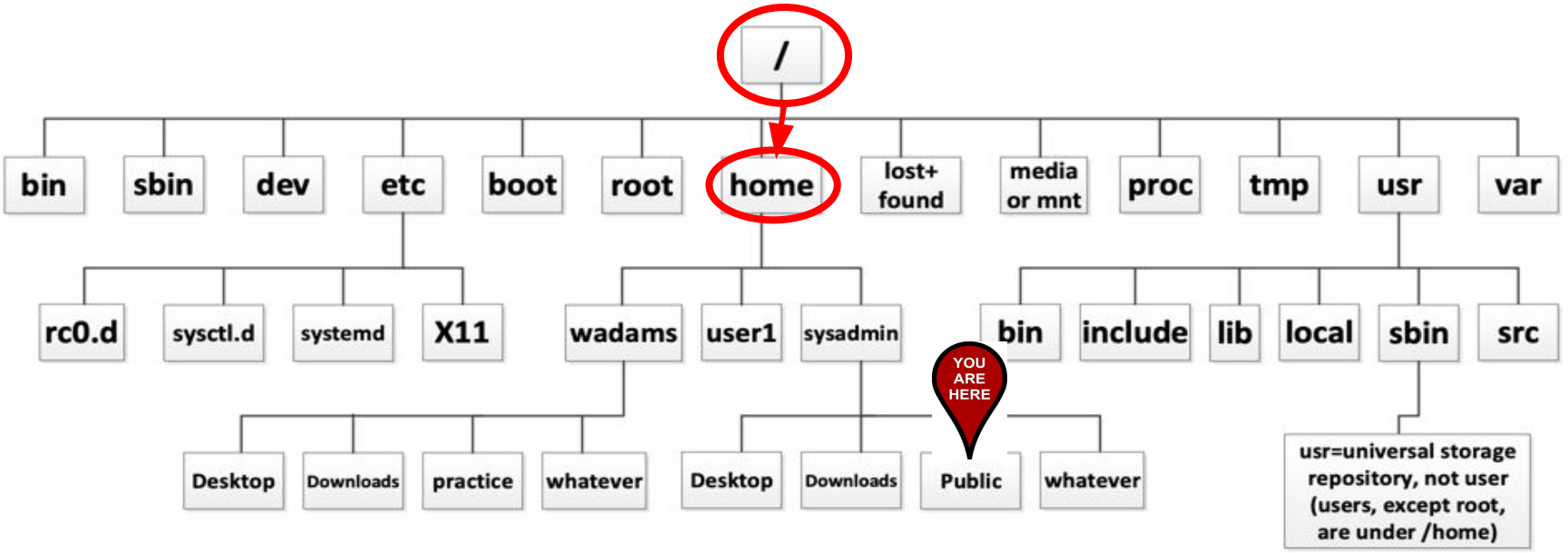
/home/wadams

Path = Location



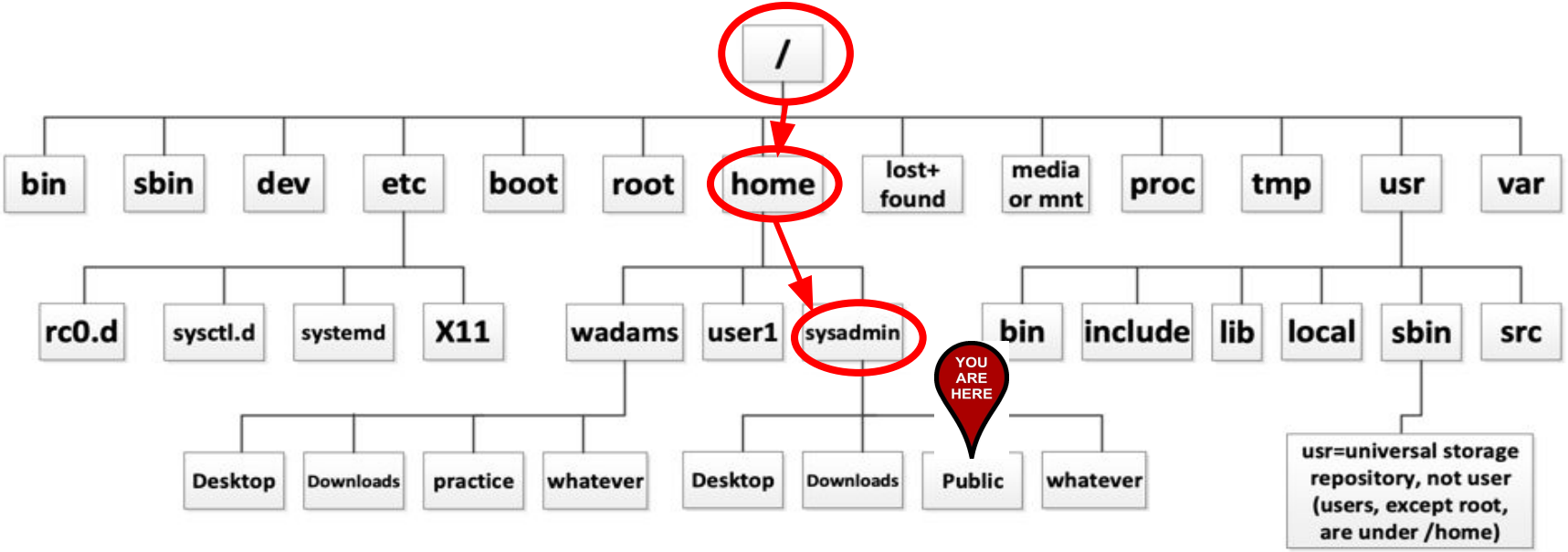
/

Path = Location



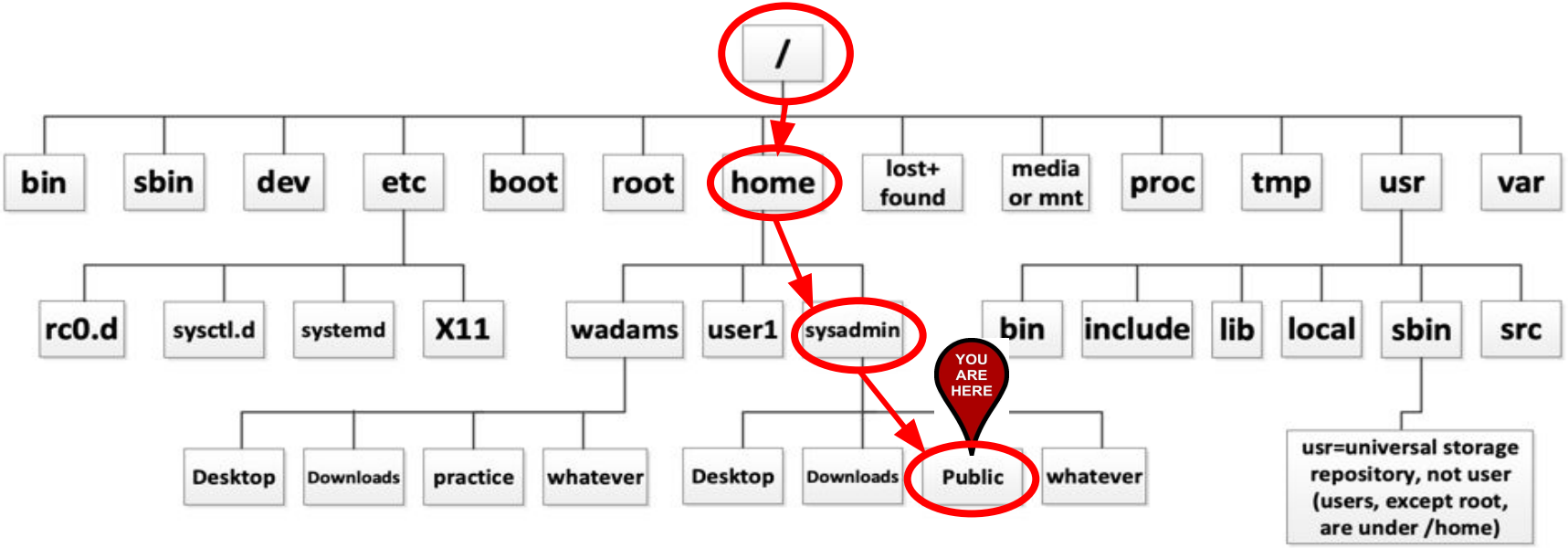
/home

Path = Location



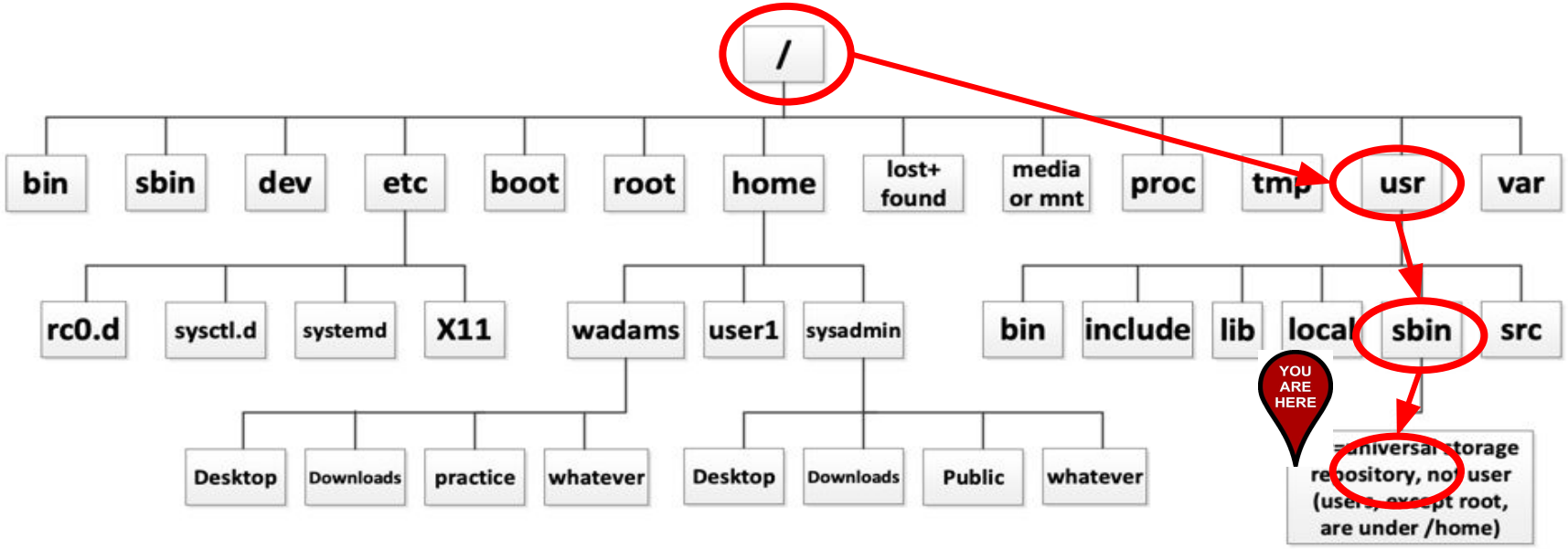
/home/sysadmin

Path = Location



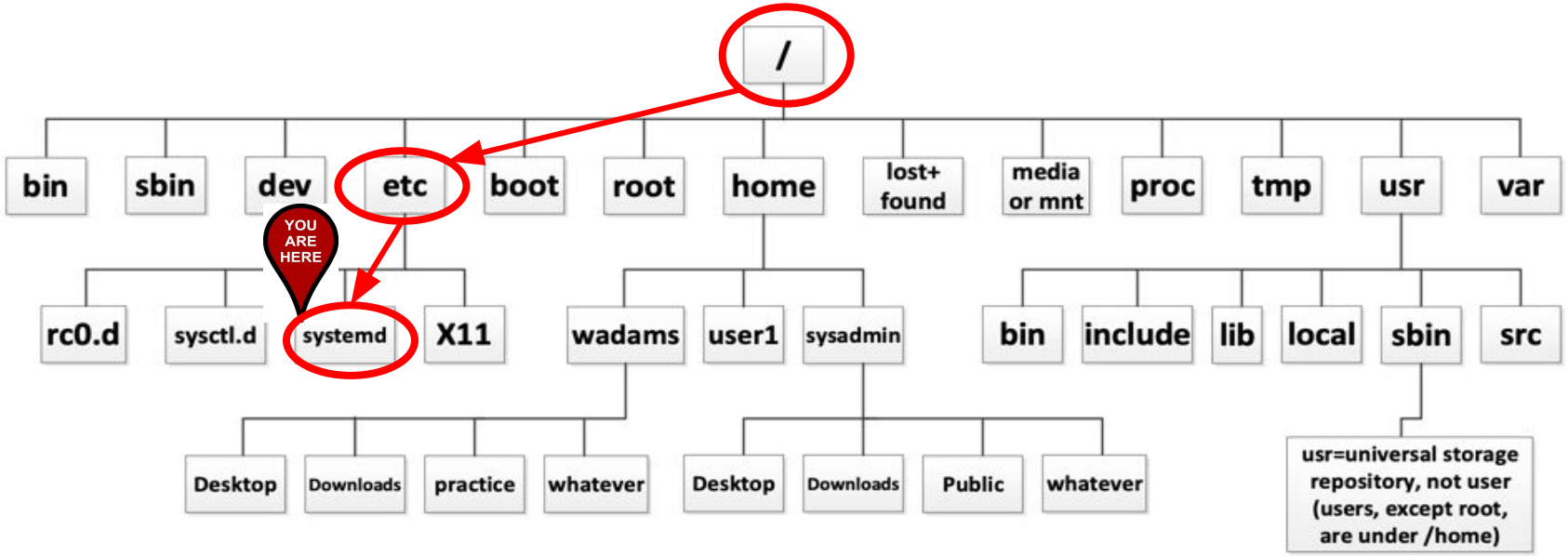
/home/sysadmin/Public

Path = Location



/usr/sbin/something

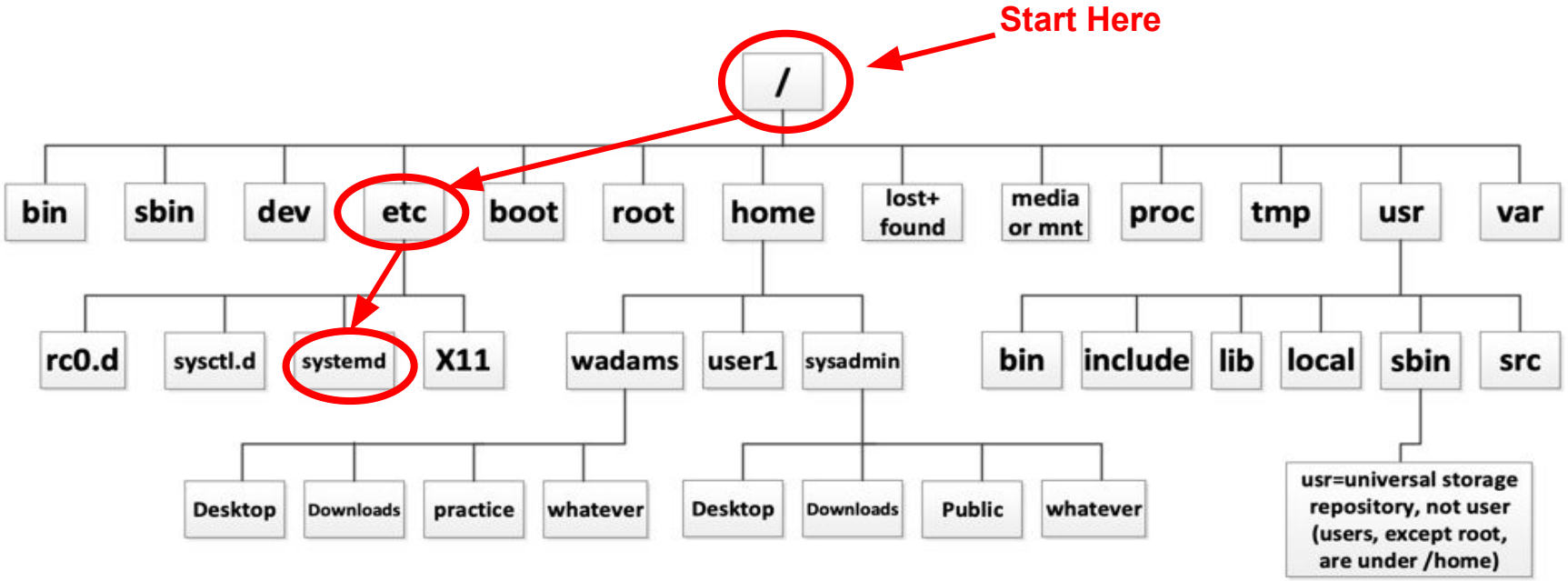
Path = Location



/etc/systemd

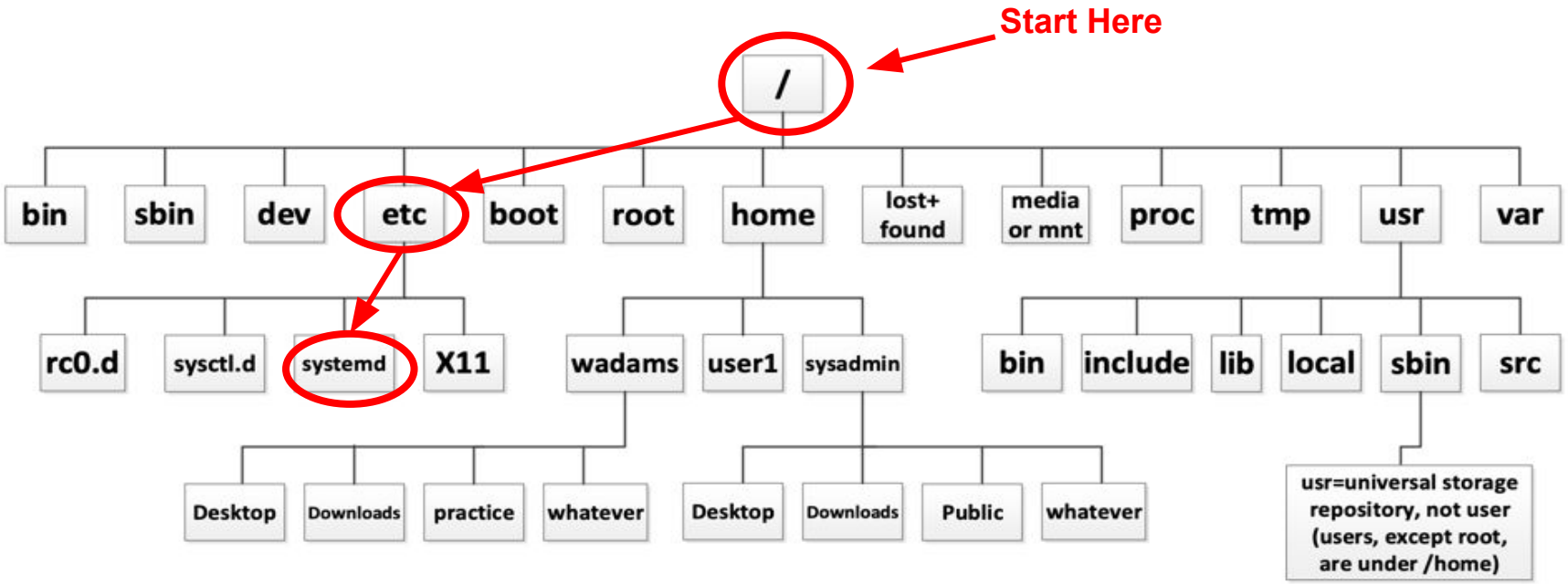
Absolute path?

Absolute Path Always Starts at Root



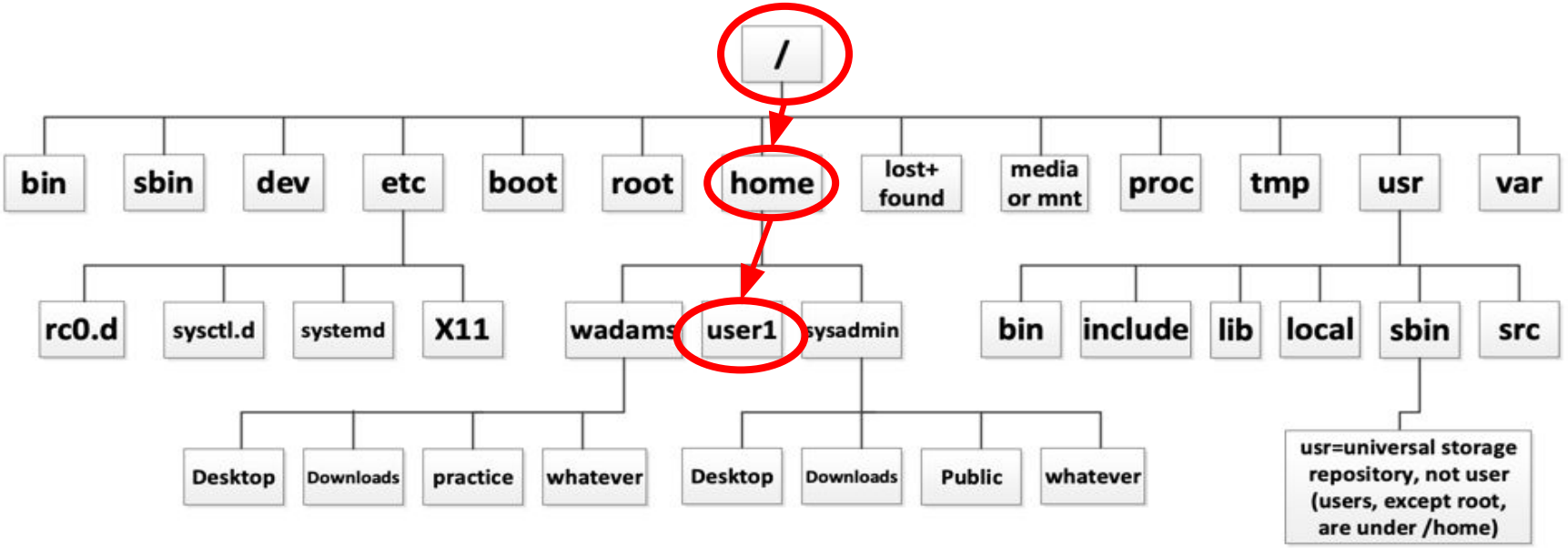
/etc/systemd

Absolute Path = Full Location of a File or Directory



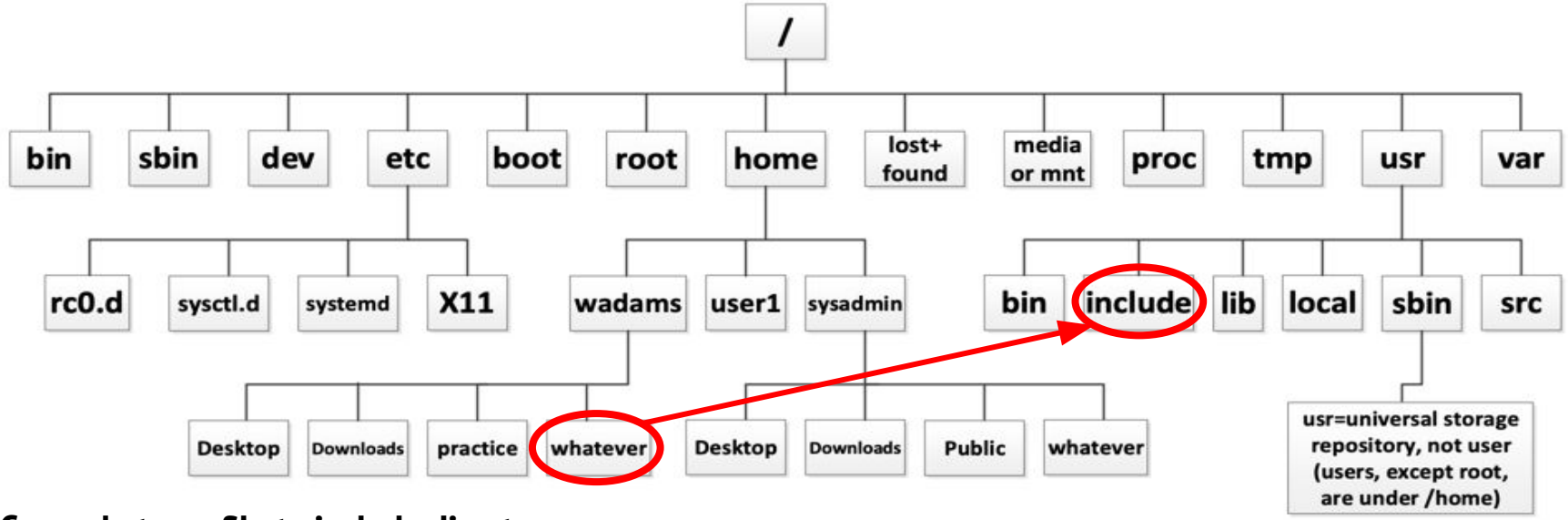
/etc/systemd

We use paths with commands - ls command



List files in /home/user1 path ---> ls /home/user1

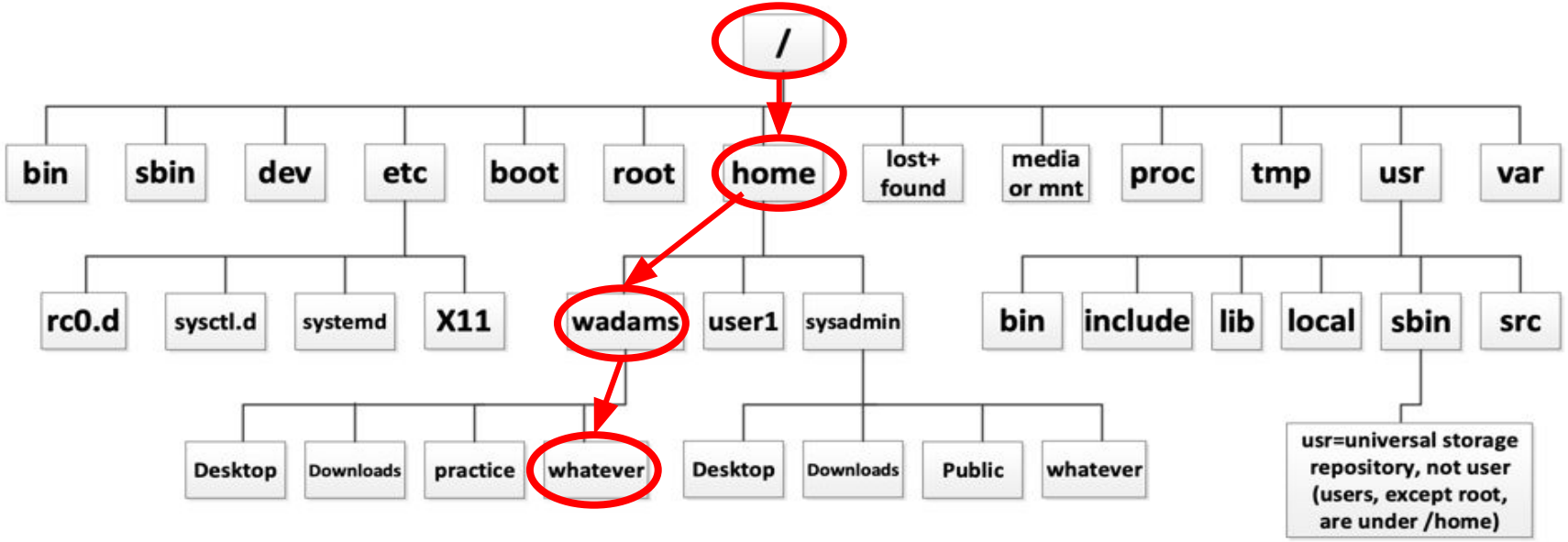
We use paths with commands - cp command



Copy whatever file to include directory

cp /home/wadams/whatever /usr/include

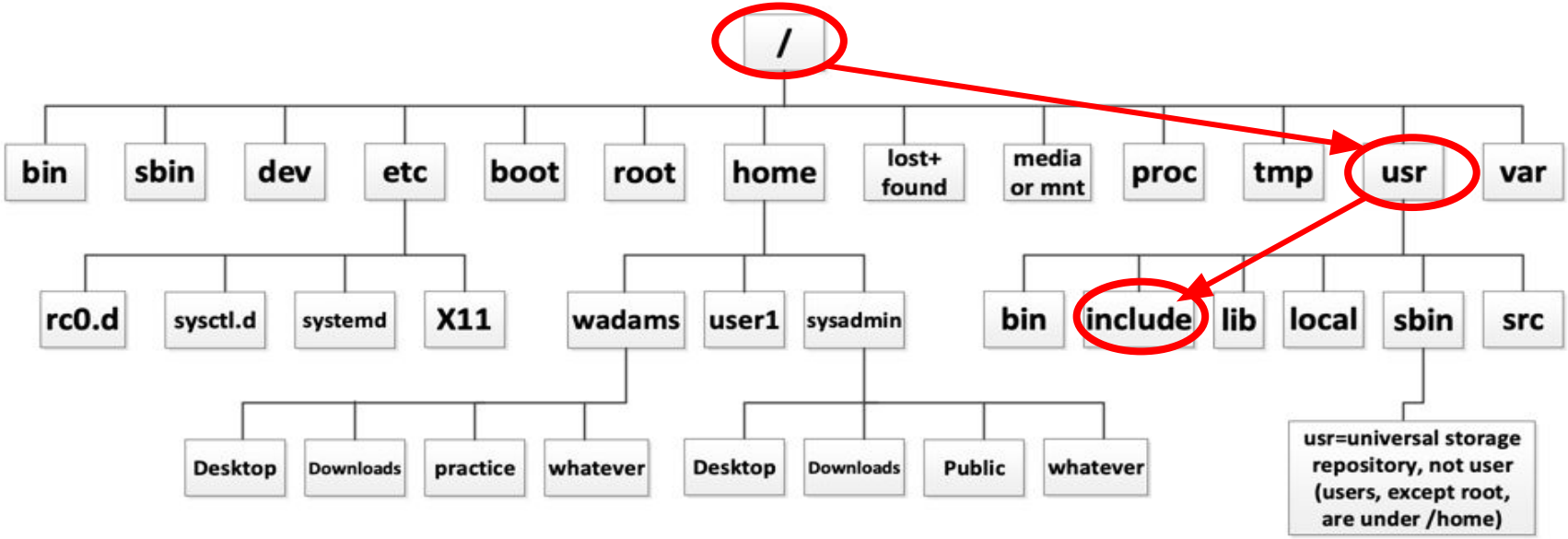
Path to whatever file - cp command



This part of the cp command indicates the path of what needs to be copied.

cp /home/wadams/whatever /usr/include

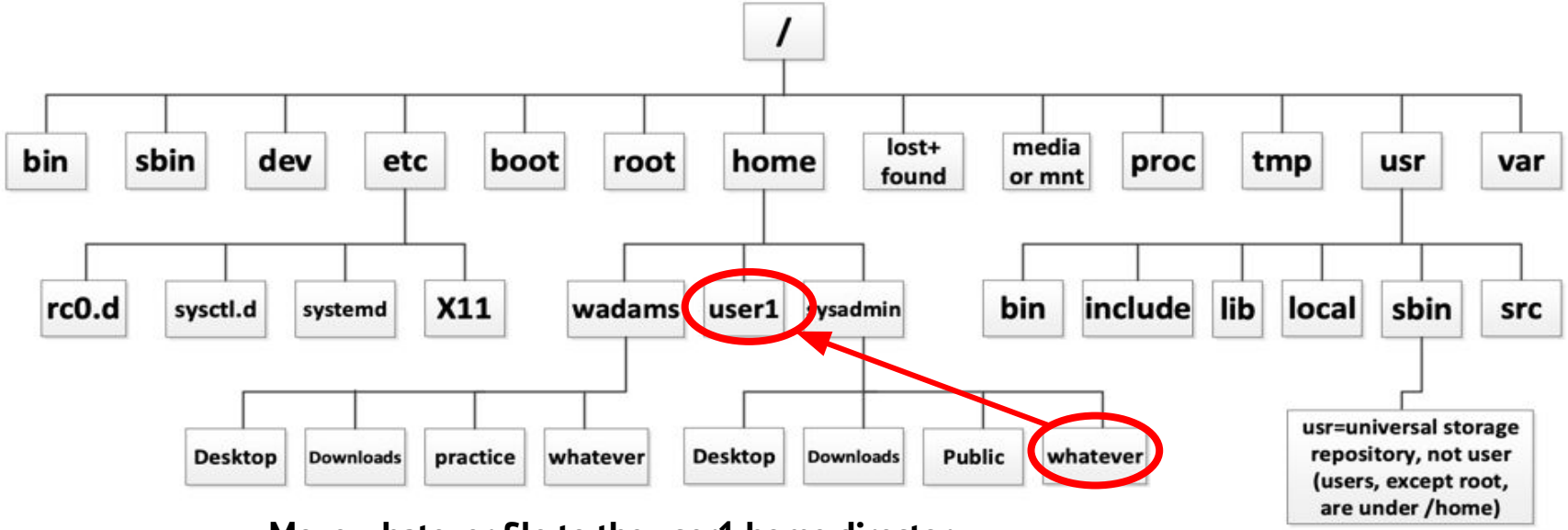
Path to include - cp command



This part of the cp command indicates the path of <u>where</u> it needs to be copied.

cp /home/wadams/whatever /usr/include

Example: Path using the mv command



Move whatever file to the user1 home directory.

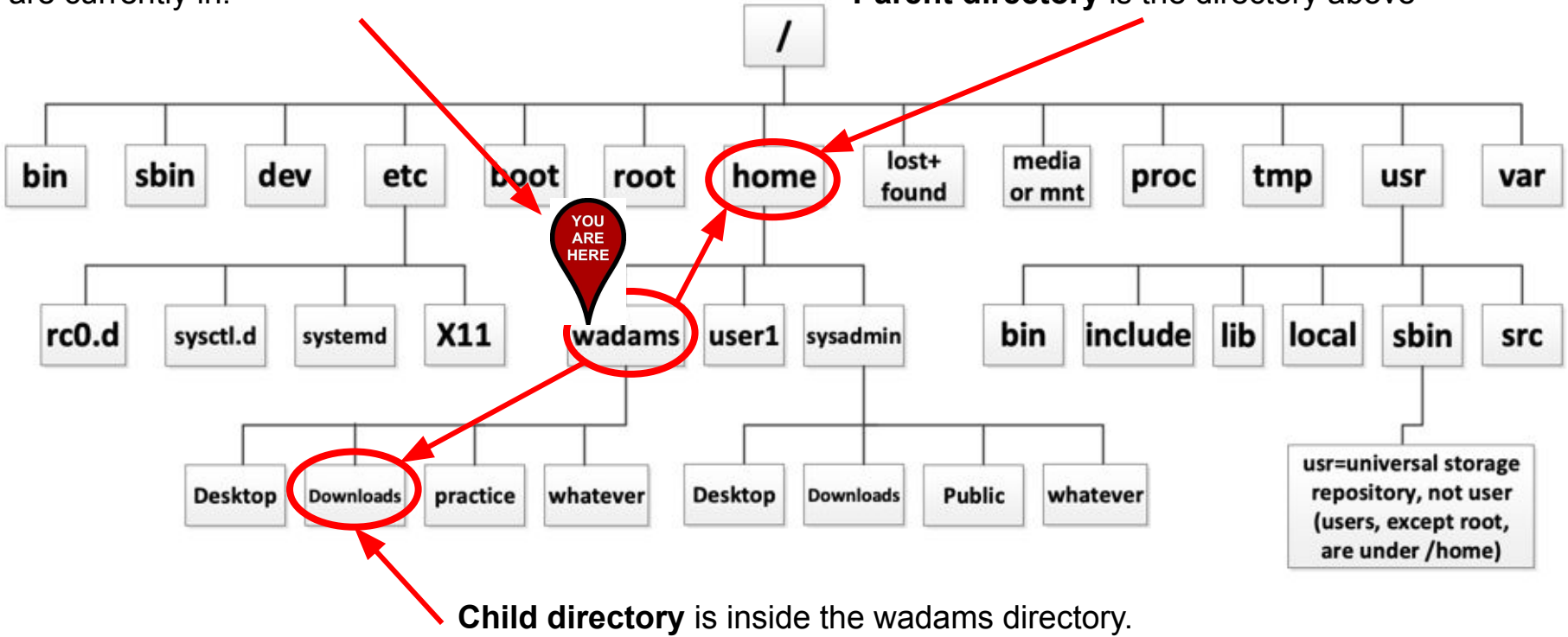
mv /home/sysadmin/whatever /home/user1

relative path?

Relative to your Current Working Directory

Current Working directory is the directory you are currently in.

Parent directory is the directory above



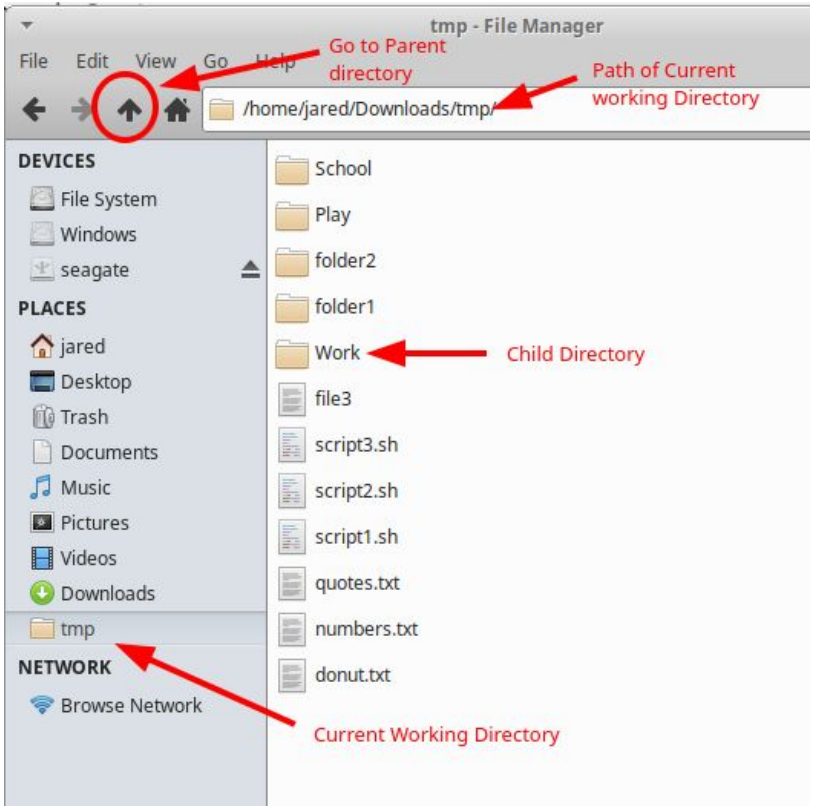
Child directory is inside the wadams directory.

Current Working Directory

Current Working Directory
- Your current location

Child Directory
- Any directory in your current directory (a sub -folder)

Parent Directory
- The directory above.

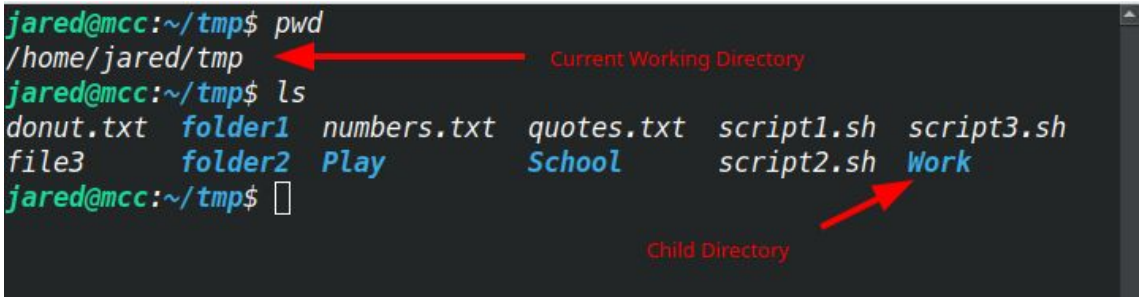


Current Working Directory

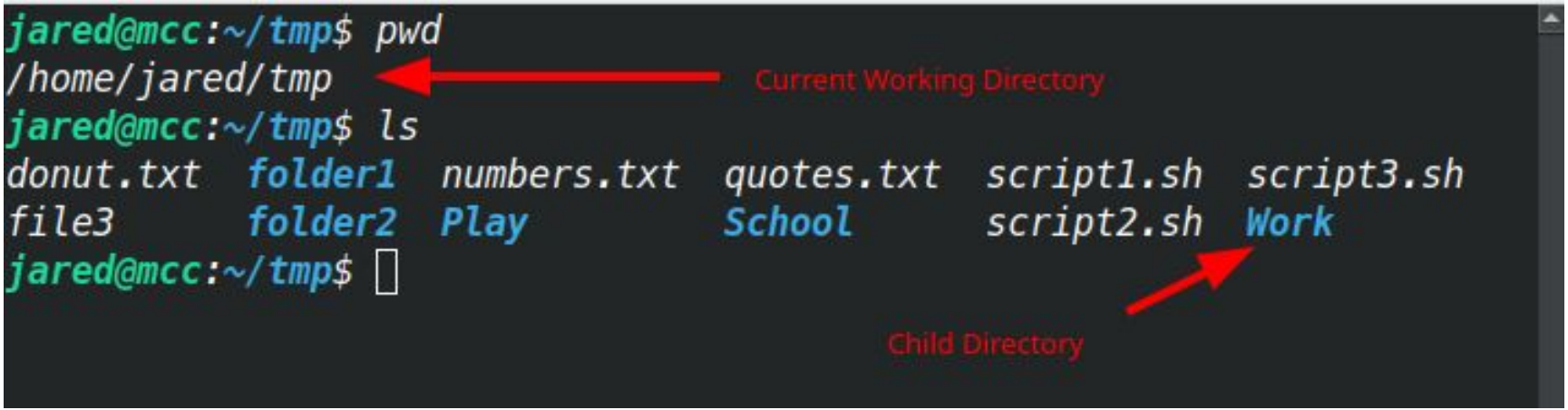
Current Working Directory
- Your current location

Child Directory
- Any directory in your current directory (a sub -folder)

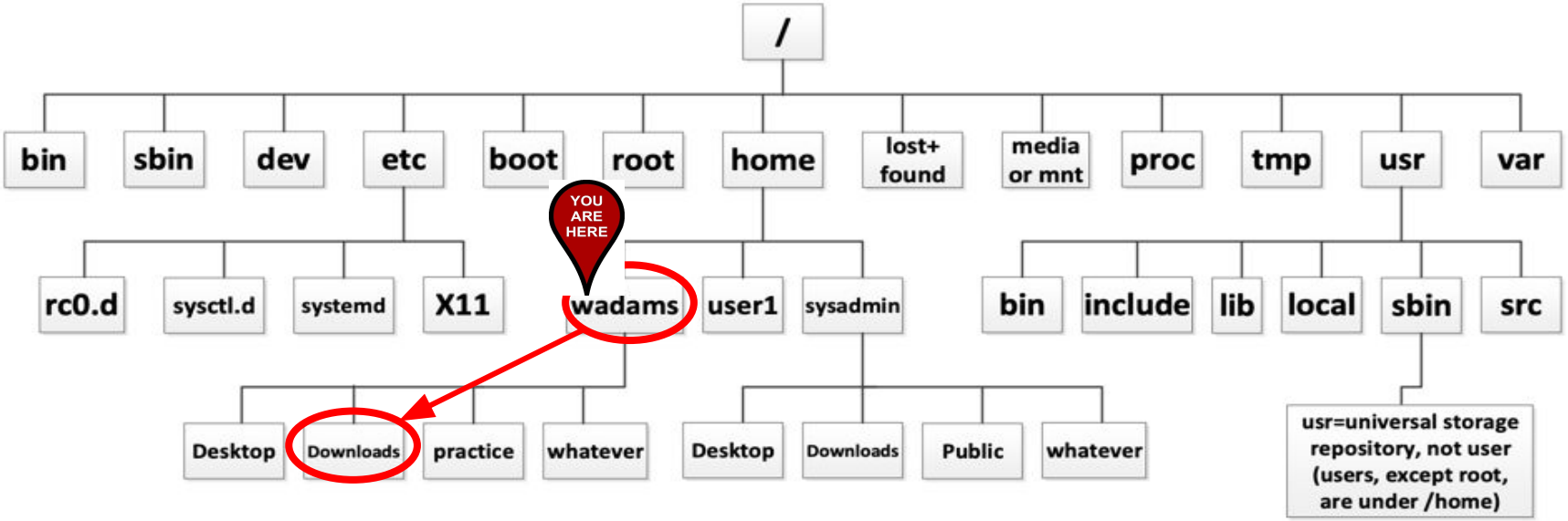
Parent Directory
- The directory above.



pwd Command - Show Current Working Directory



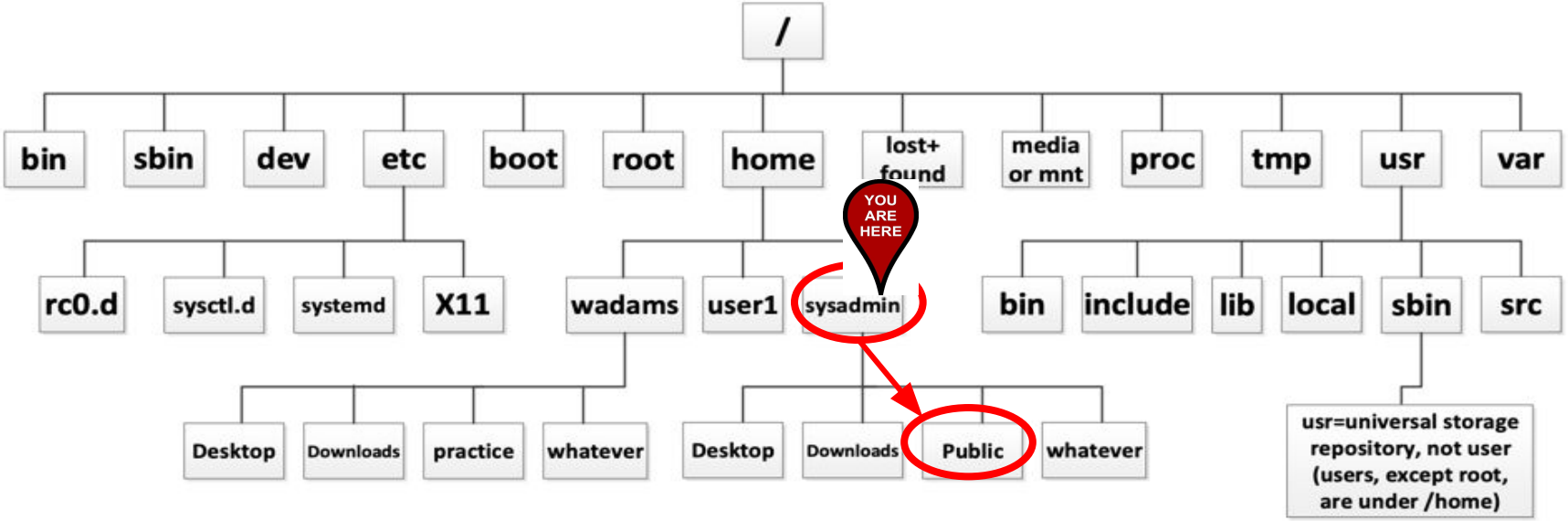
Relative Paths to Child Directories



The full absolute path is NOT needed, only a part of the path, *relative* to your current work directory.

Downloads

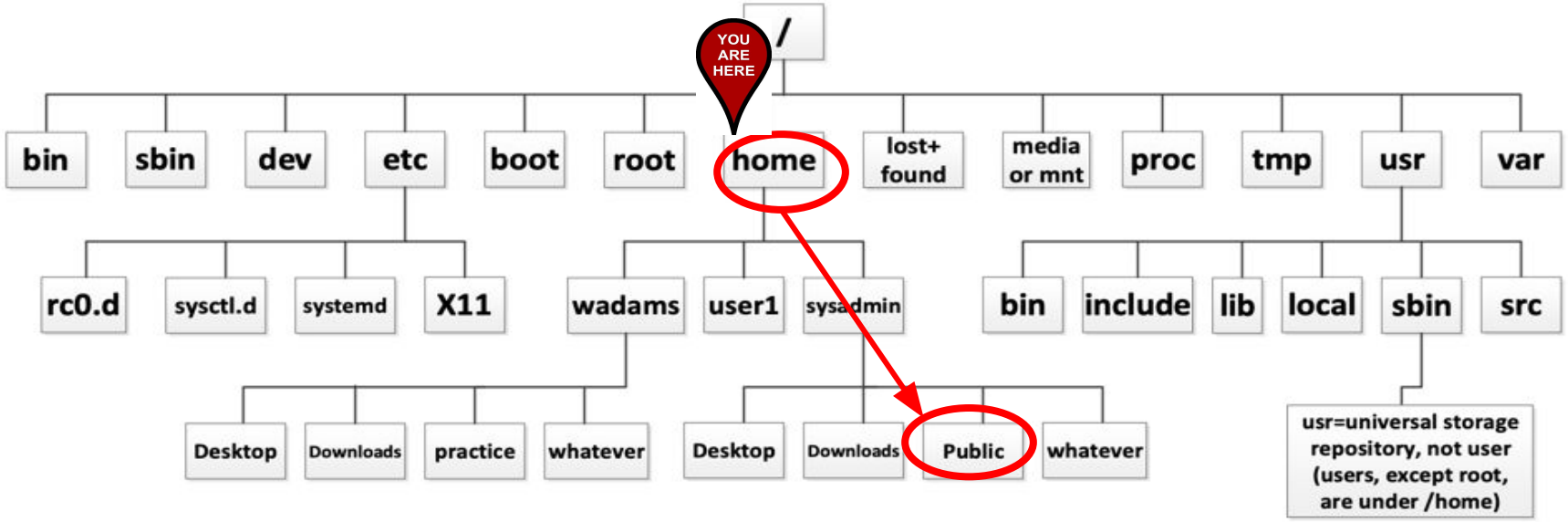
Relative Paths to Child Directories - Command Examples



Using a relative path to view the child directory Public.

ls Public

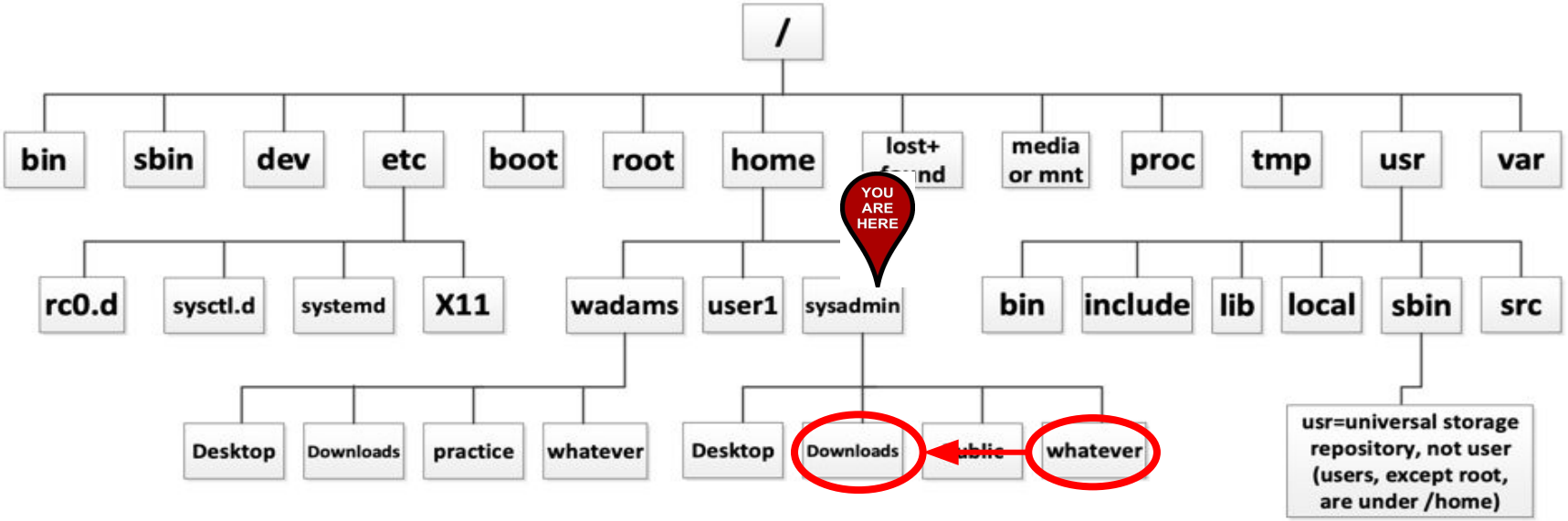
Relative Paths to Child Directories - Command Examples



Using a relative path to view the Public folder 2 levels deep.

ls sysadmin/Public

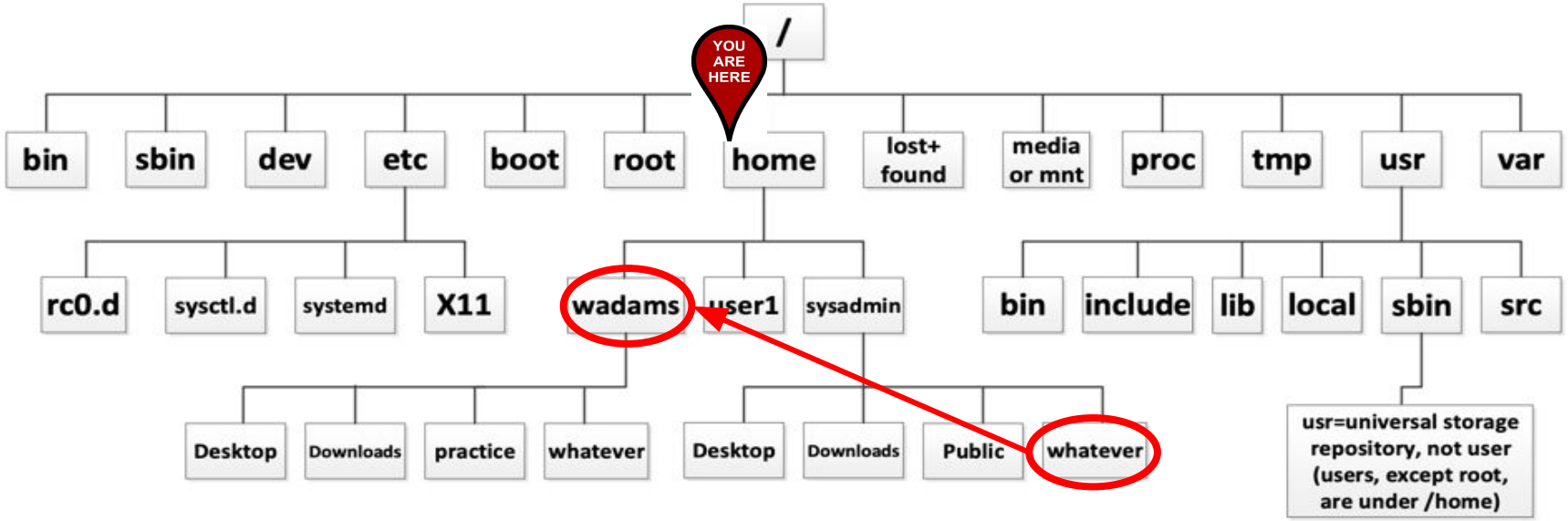
Relative Paths to Child Directories - Command Examples



Using a relative path to copy whatever to Downloads. Both whatever and Downloads are at the same child level.

cp whatever Downloads

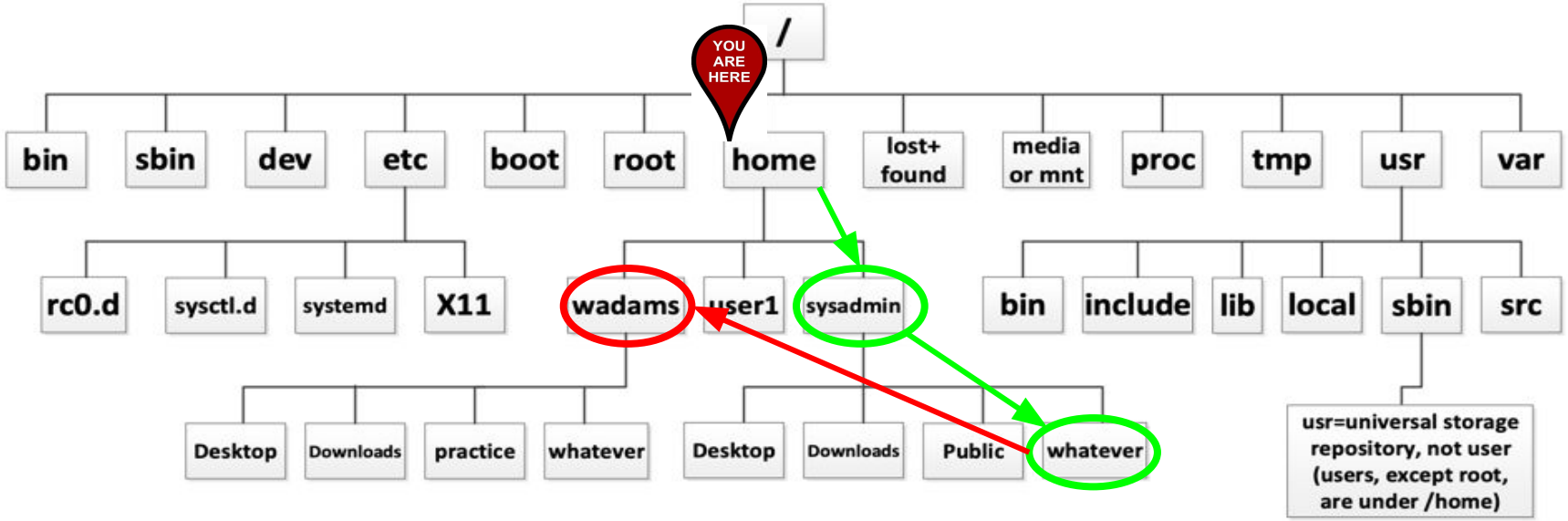
Relative Paths to Child Directories - Command Examples



Using a relative path to move whatever to wadams. whatever is 2 child levels down.

mv sysadmin/whatever wadams

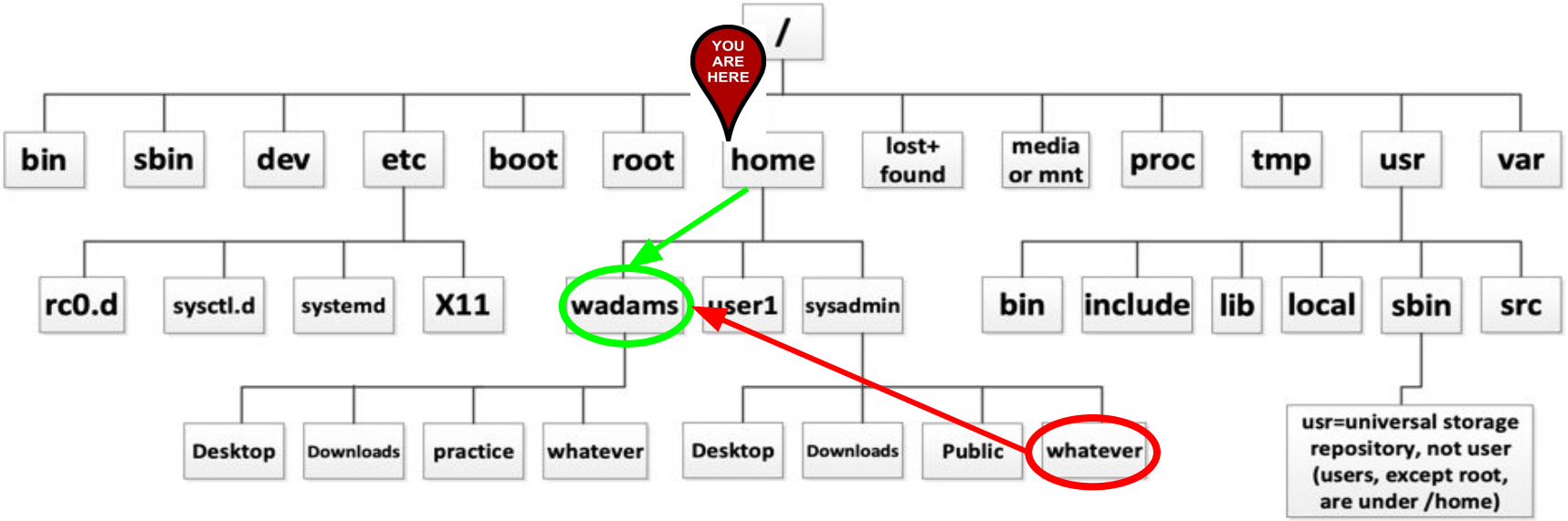
Relative Paths to Child Directories - Command Examples



Relative path of what you want to move.

mv sysadmin/whatever wadams

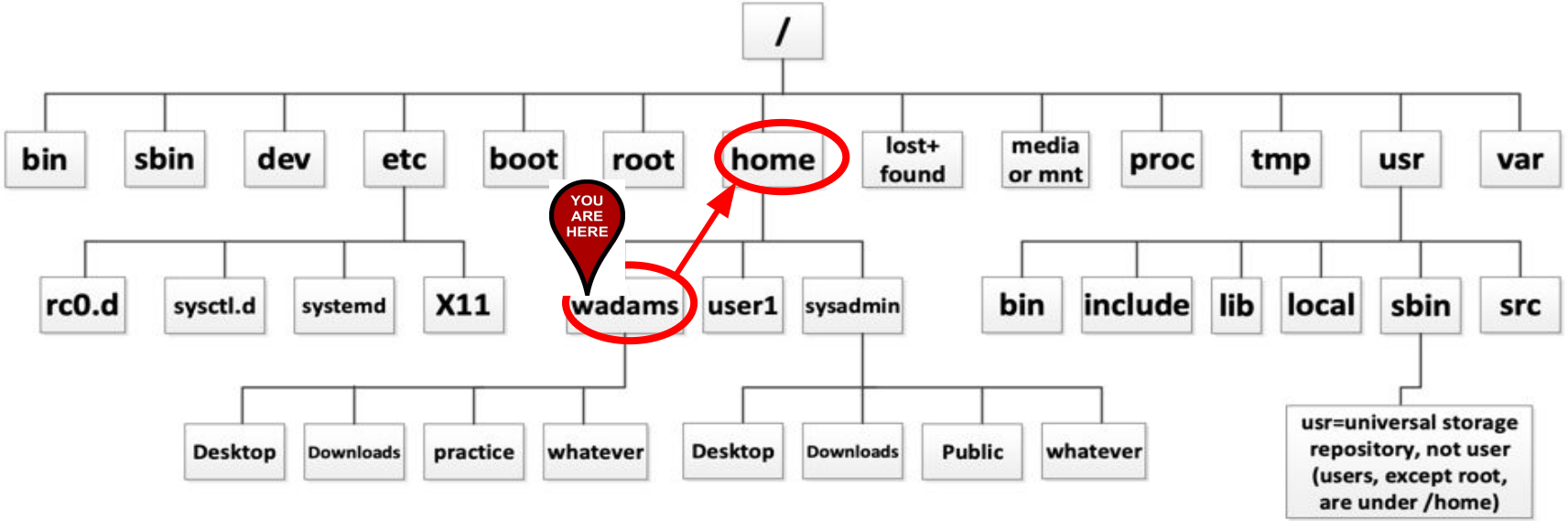
Relative Paths to Child Directories - Command Examples



Relative path of where you want to move it.

mv sysadmin/whatever wadams

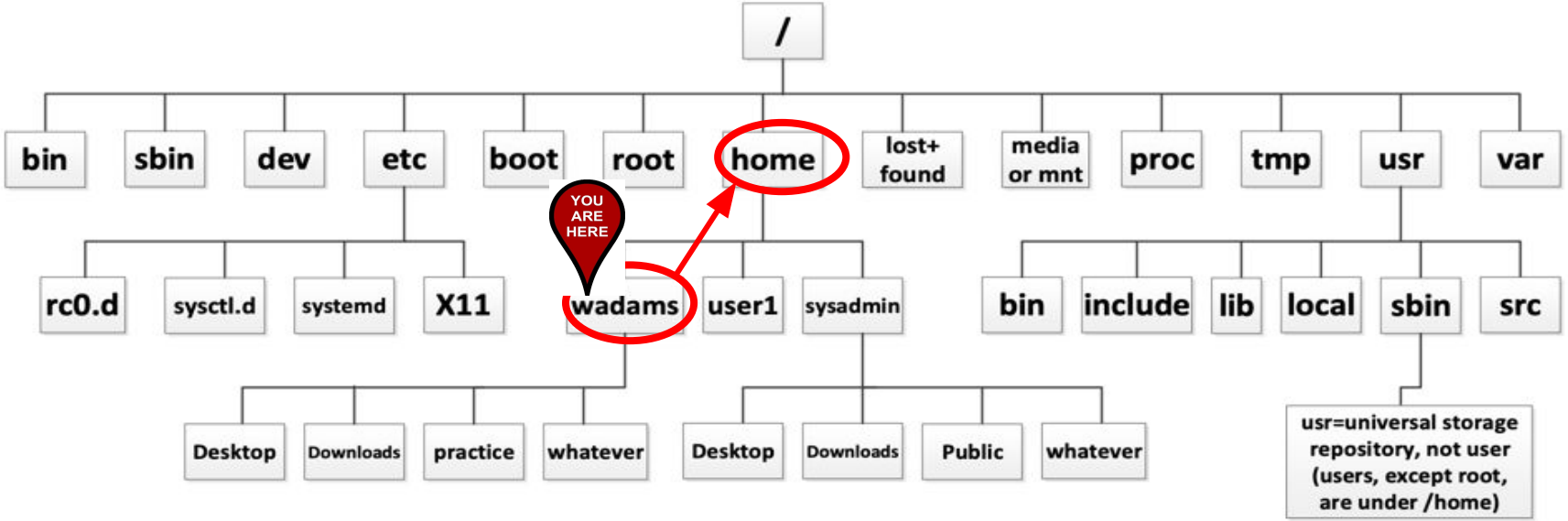
Relative Paths to Parent Directories



When referring to parent directories use .. (2 dots).

..

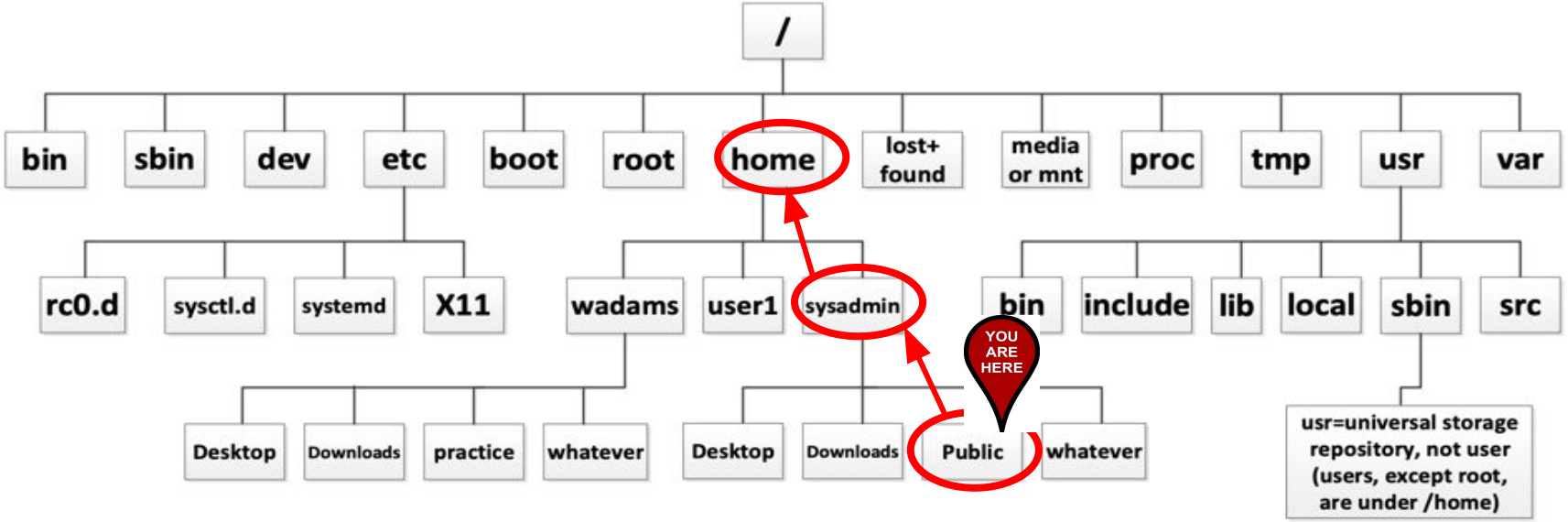
Relative Paths to Parent Directories - Example command



Using the relative path to view /home.

ls ..

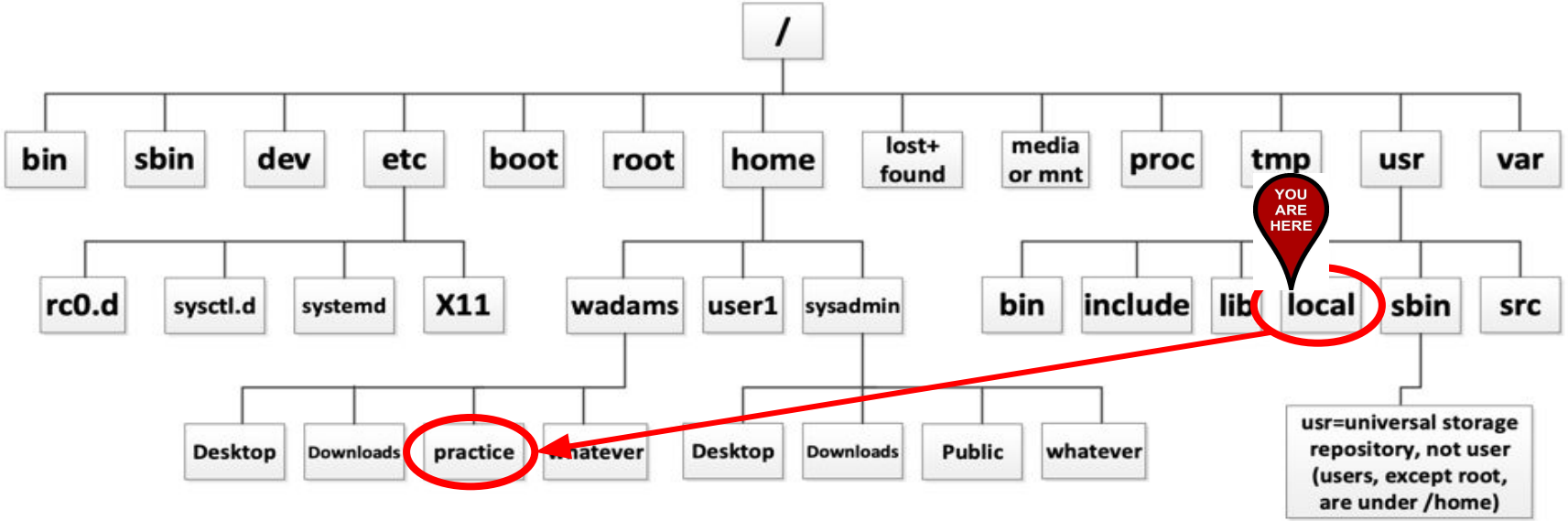
Relative Paths to Parent Directories - Example command



Using the relative path to view up 2 levels. Use a / when moving up to the next level.

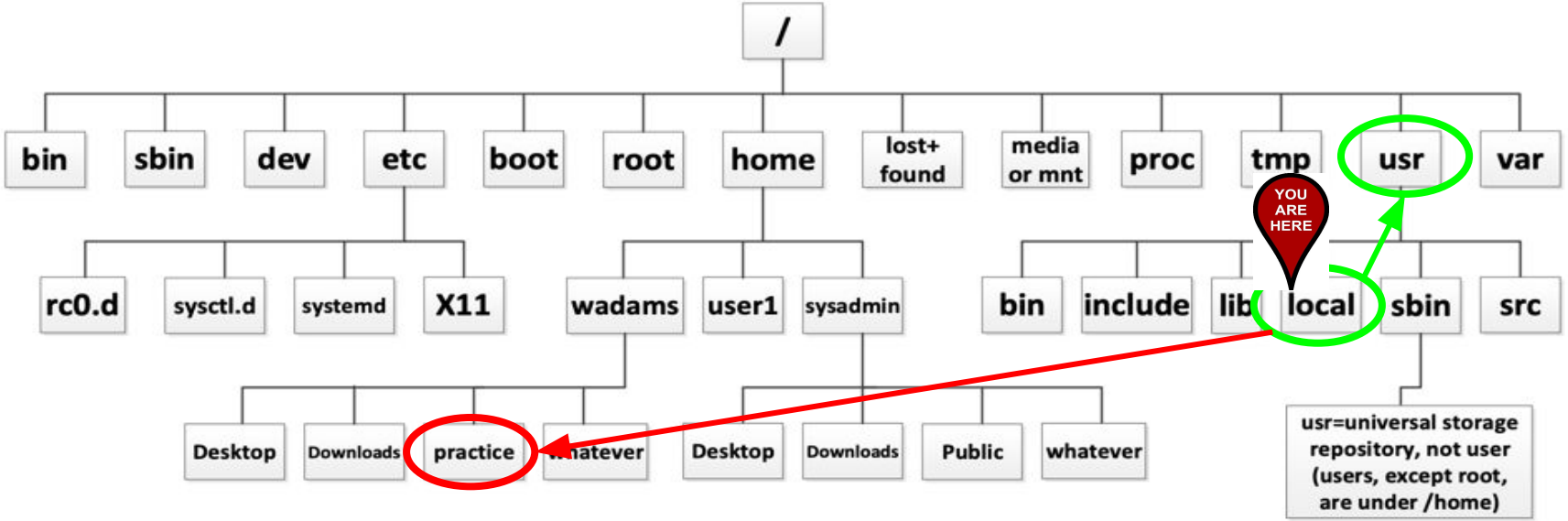
ls ../..

Relative Paths



How to use a relative path to view /home/wadams/practice when my current working directory is /usr/local.

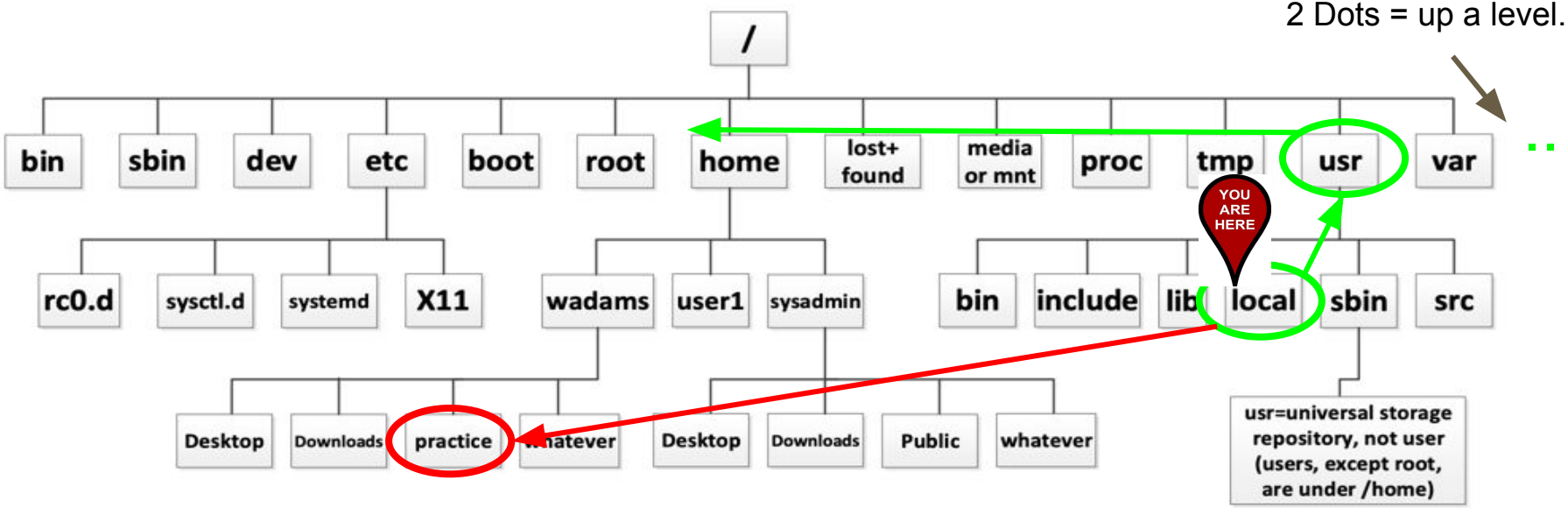
Relative Paths



Up to the parent directory, use .. (2 dots).

```
ls ..
```

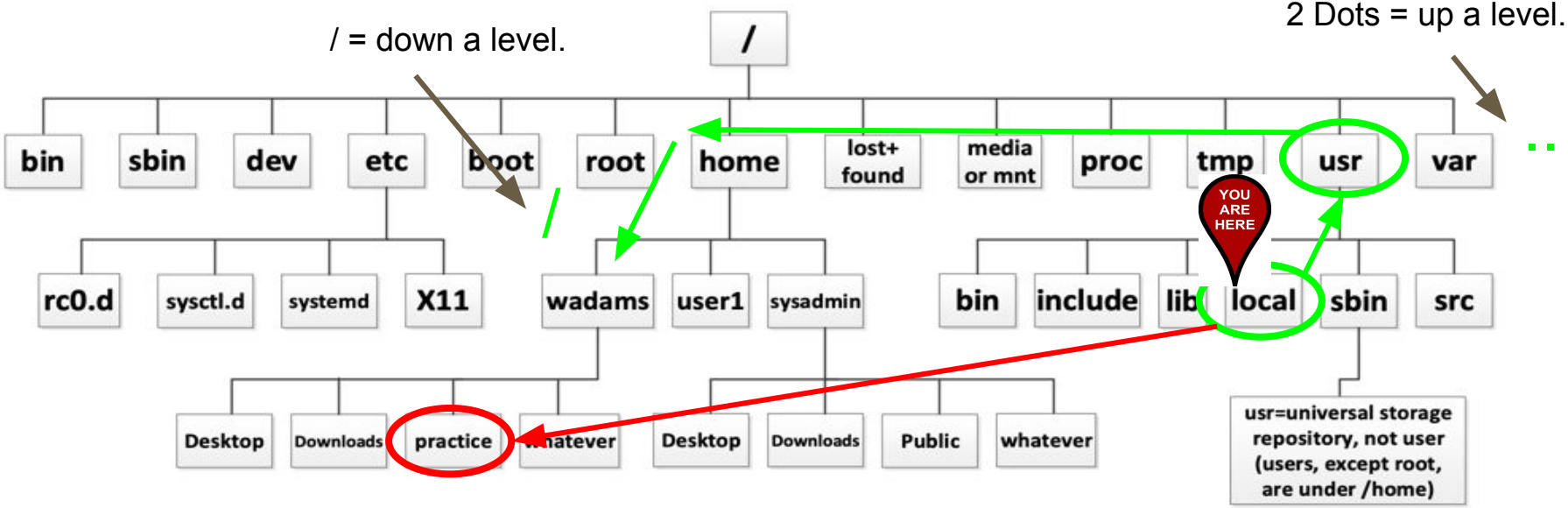
Relative Paths



The home directory is on the same level as usr.

ls ..

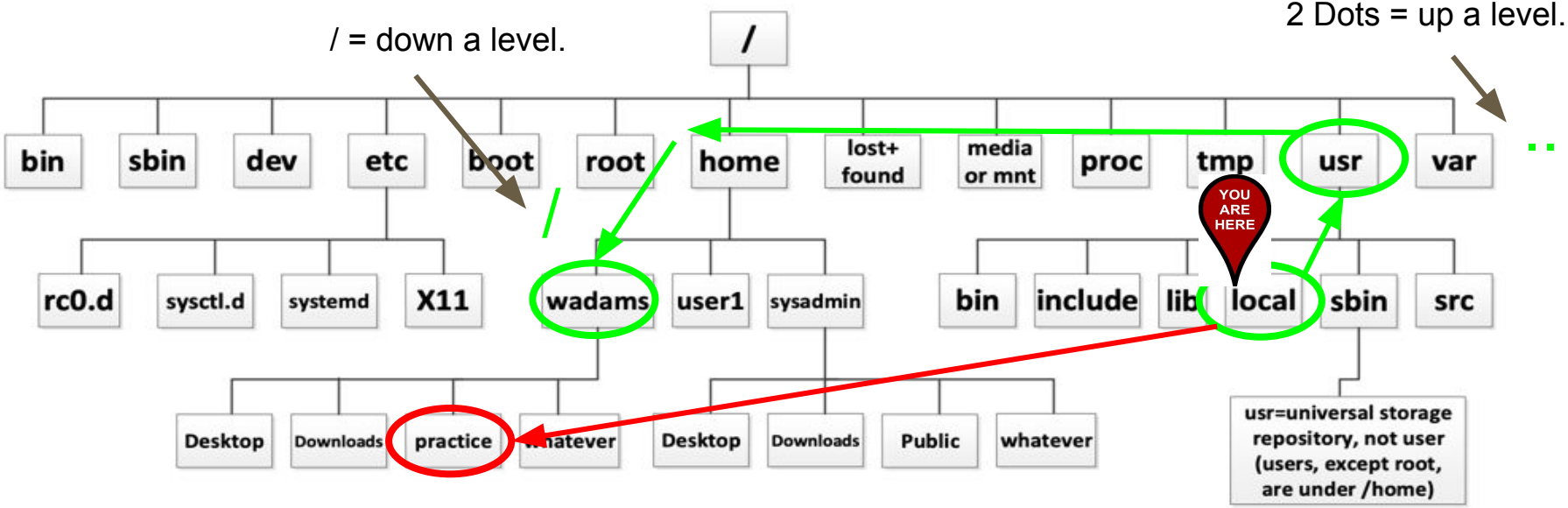
Relative Paths



Add a slash when moving down a level.

ls ../

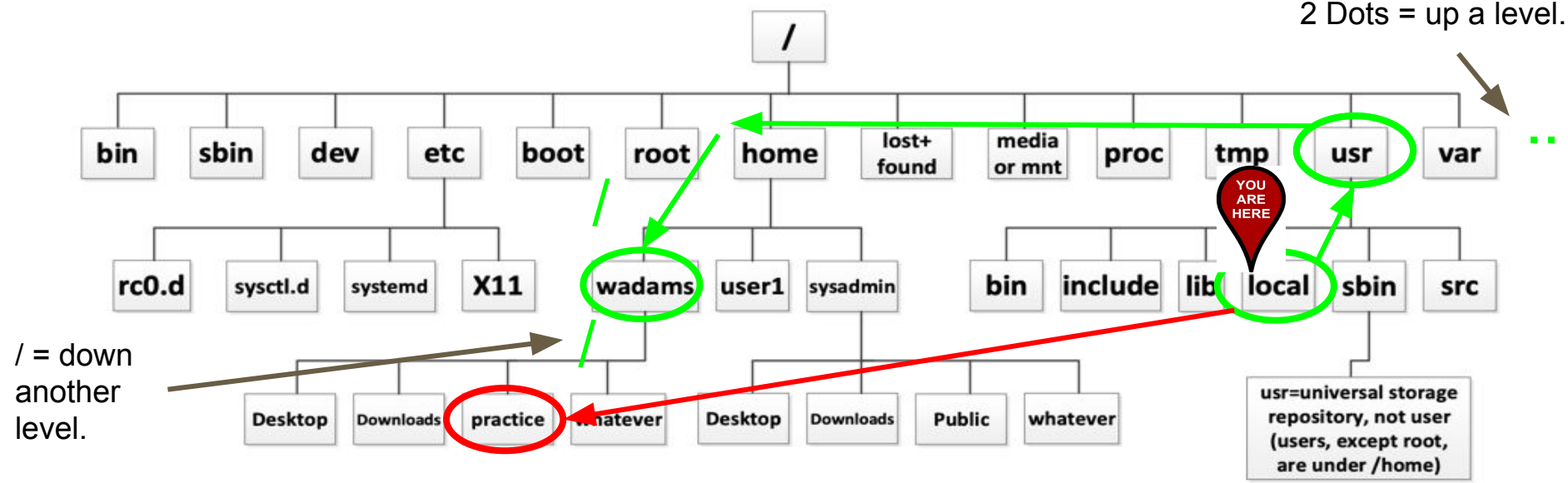
Relative Paths



Add the name of the directory.

ls ../wadams

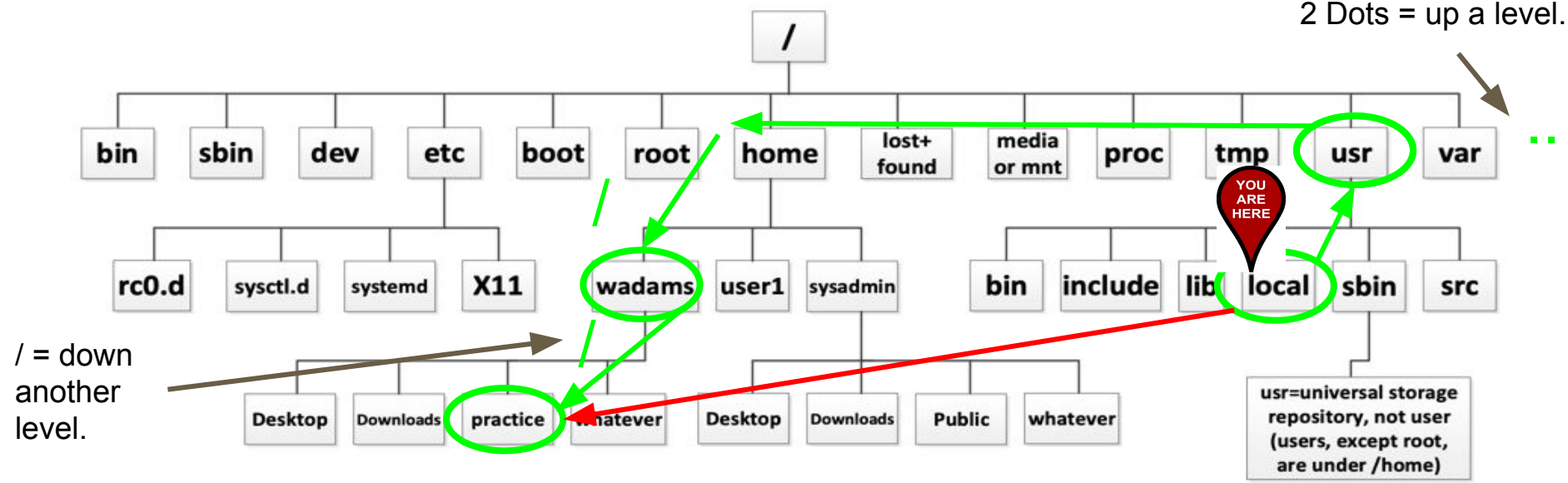
Relative Paths



Moving down to another level, so add another / (forward slash).

ls ../wadams/

Relative Paths



Finally, add the name of the destination directory.

ls ../wadams/practice

Relative Paths Summary

Up 1 level	..
Up 2 levels	../..
Up 3 levels	../../..
Up 1 level, down 1 level	../dir1
Up 2 levels, down 1 level	../../dir1
Up 2 levels, down 2 levels	../../dir1/dir2
Up 2 levels, down 3 levels	../../dir1/dir2/dir3
Down 1 level	dir1
Down 2 levels	dir1/dir2
Down 3 levels	dir1/dir2/dir3

Up the filesystem use:

.. (2 dots)

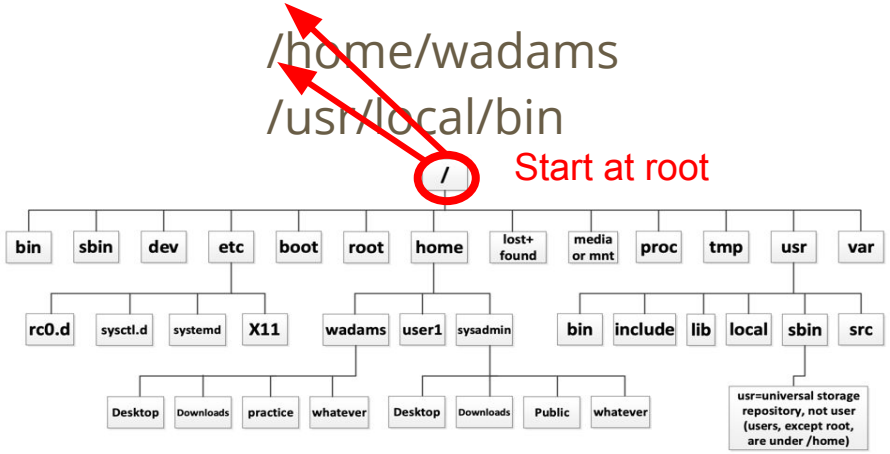
Down the filesystem:

Use the name of the directory

Use a / (forward slash) to designation a new level.

Absolute and Relative Paths Summary

Absolute paths always begin with a / (forward slash) because they start at root (/)



- **Relative** paths **NEVER** begin with a / (forward slash).
- They **only** begin with .. (2 dots) **or** the name of a directory.
- They are relative to your current working directory.

../../..
dir1
../../dir1/dir2