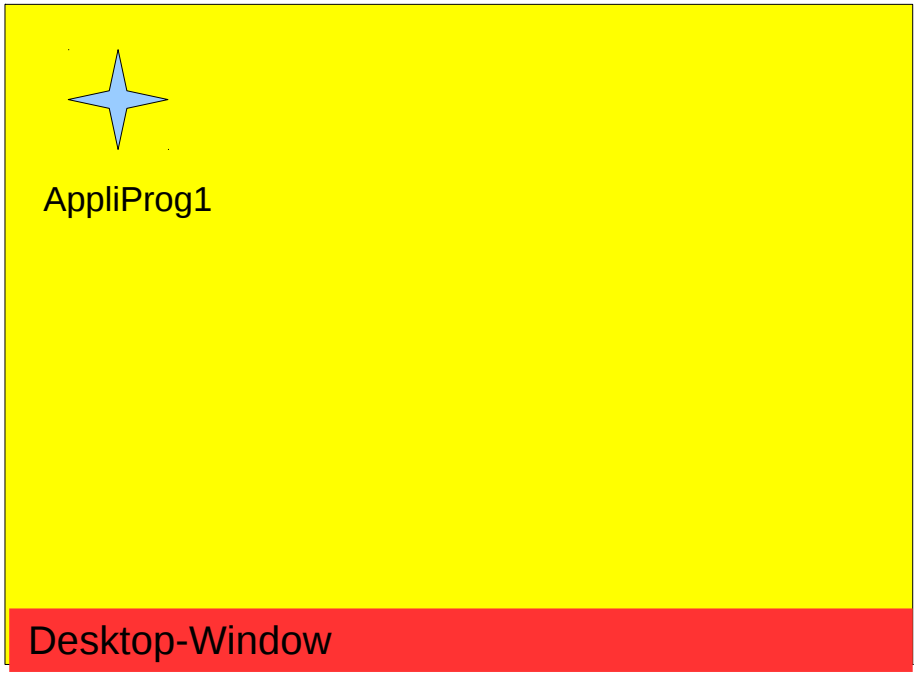
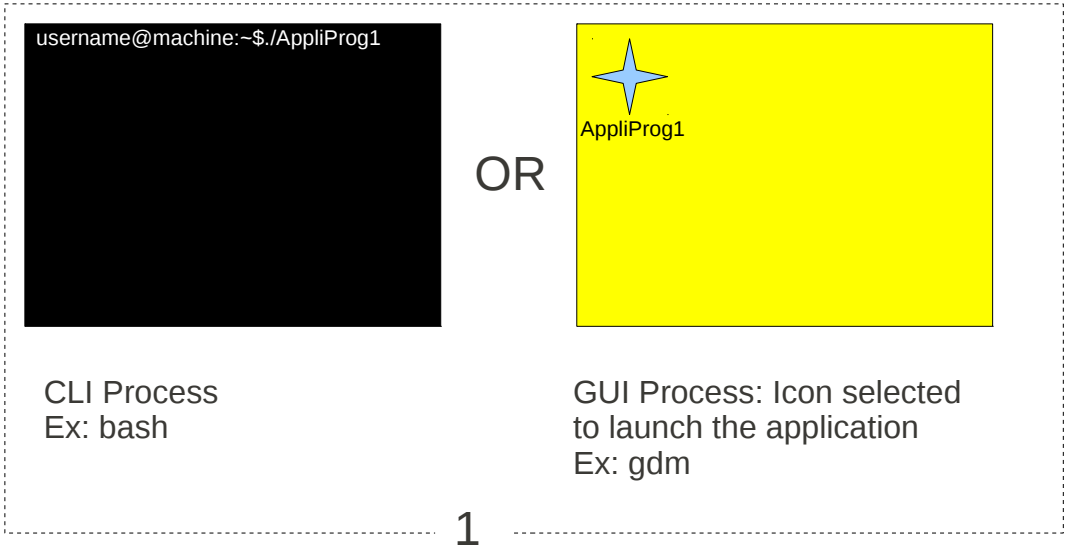
OR

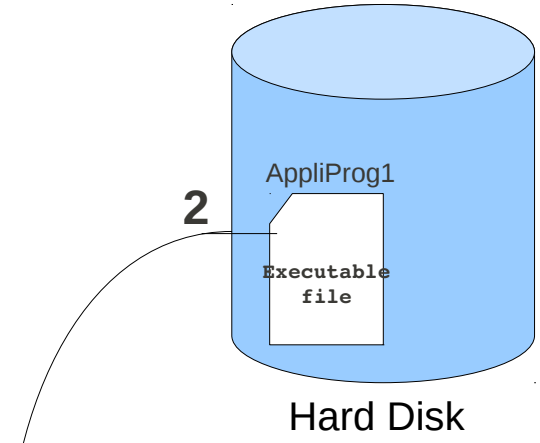


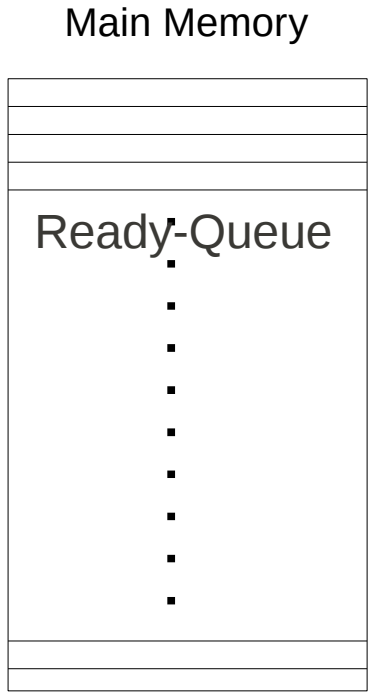
CLI Process
Ex: bash

GUI Process: Icon selected to launch the application
Ex: gdm

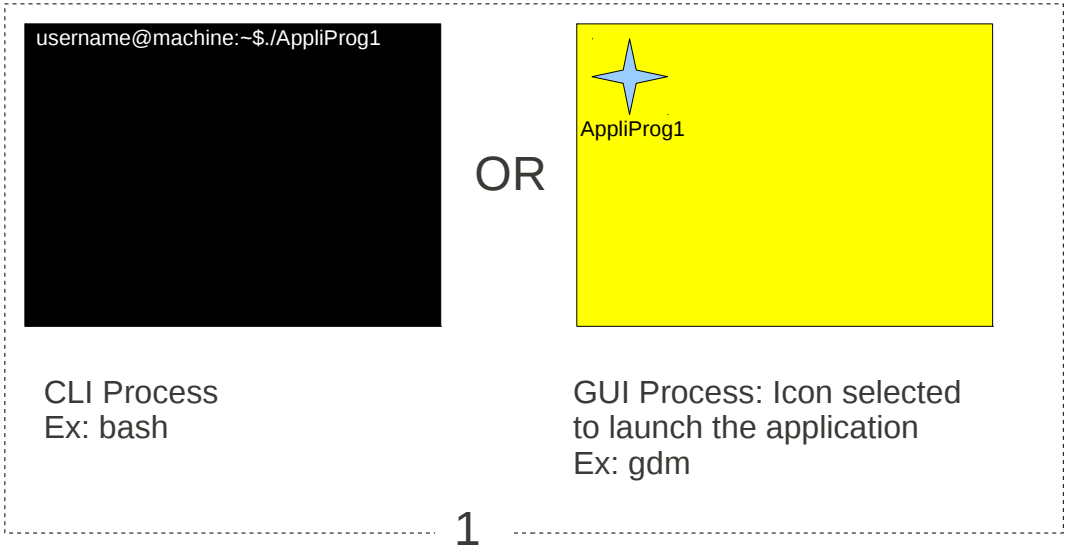
1. User launches “AppliProg1” via CLI or GUI

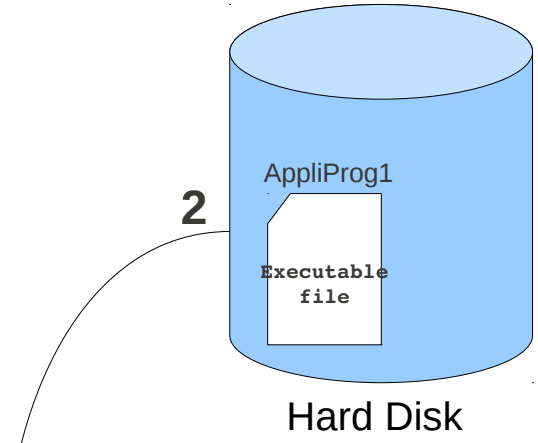


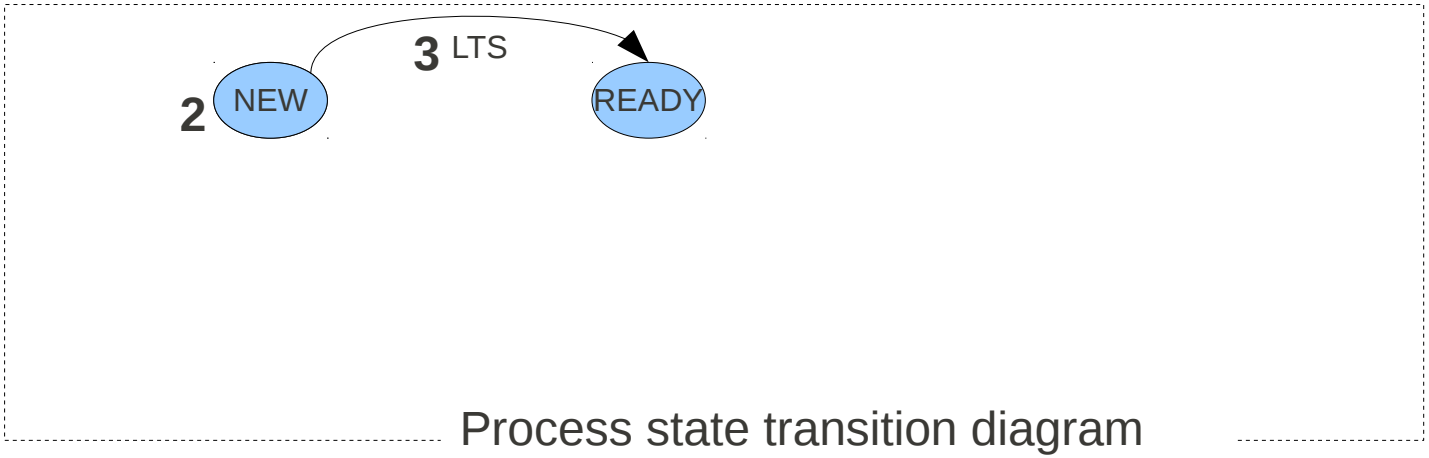


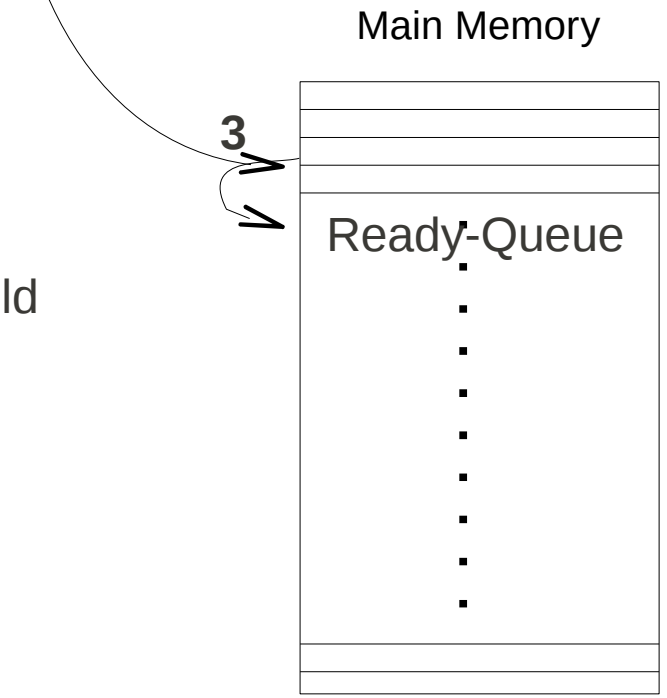


1. User launches “AppliProg1” either by CLI/GUI/...
2. The CLI/GUI process **fork/clones** to execute AppliProg1 as its child
Now this process is in **NEW state.** Here prog loads into memory and process gets its PCB

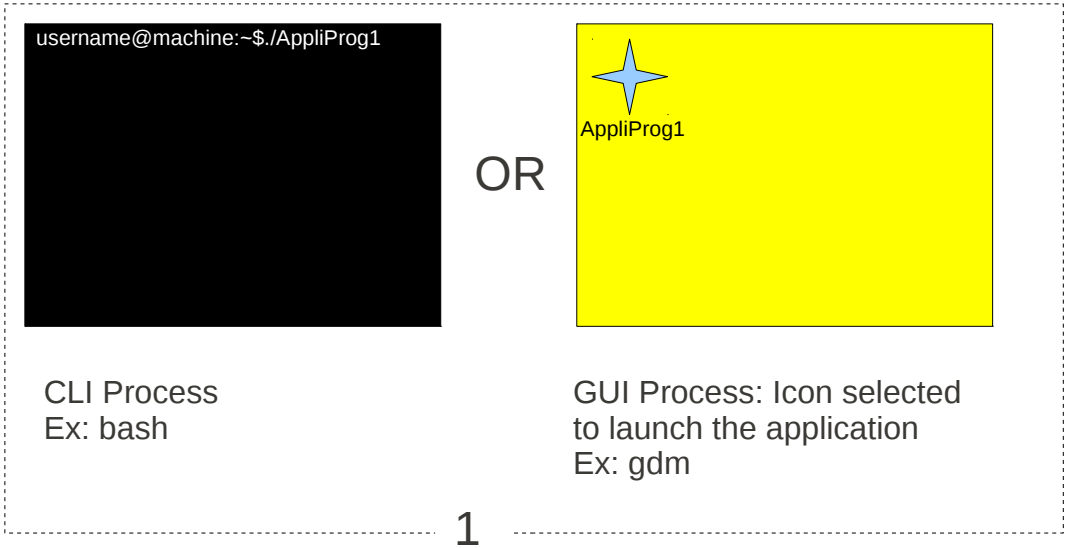


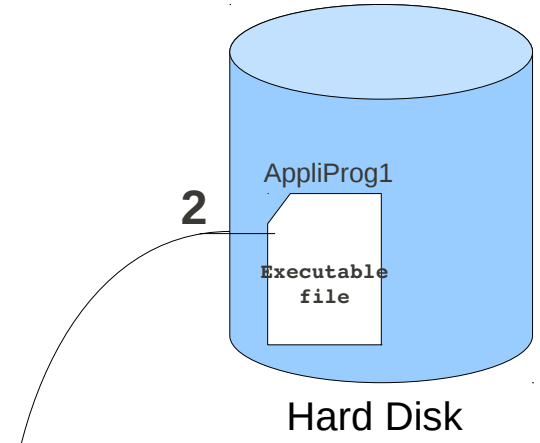


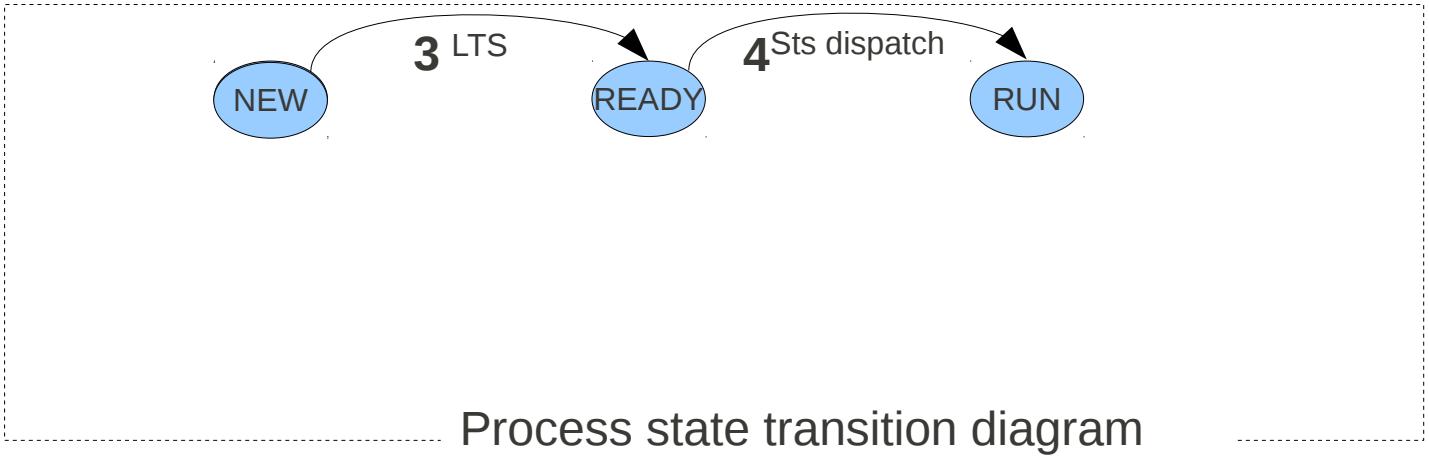


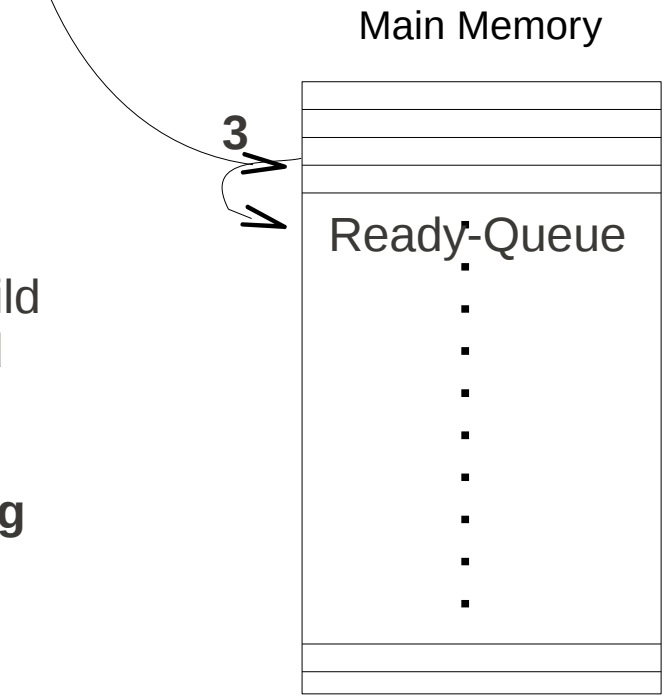


1. User launches “AppliProg1” either by CLI/GUI/...
2. The CLI/GUI process **fork/clones** to execute AppliProg1 as its child
Now this process is in **NEW state.**
3. LTS moves this proc into **Ready state** into Ready-queue

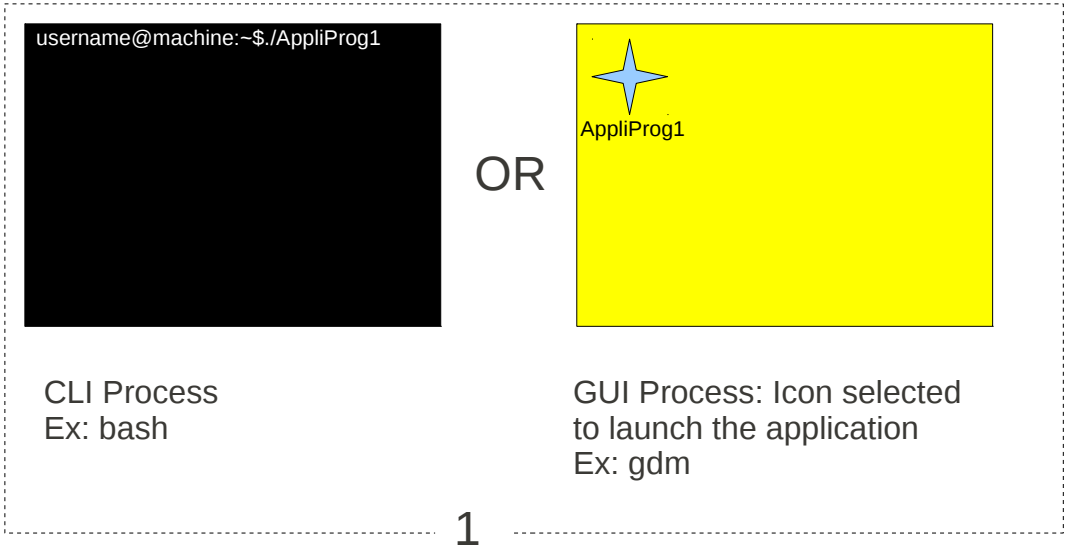


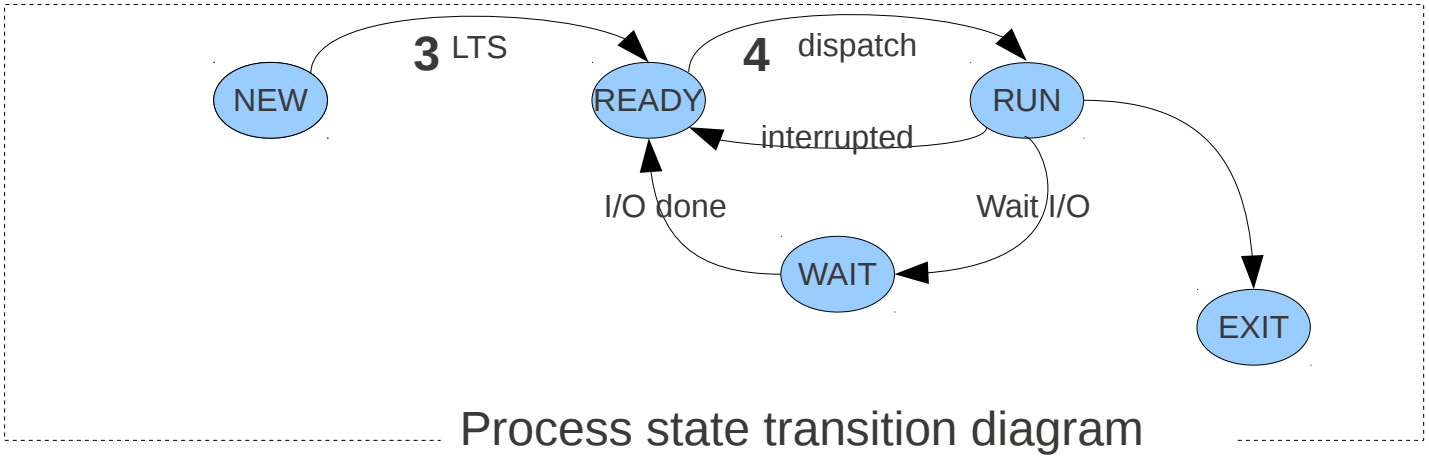


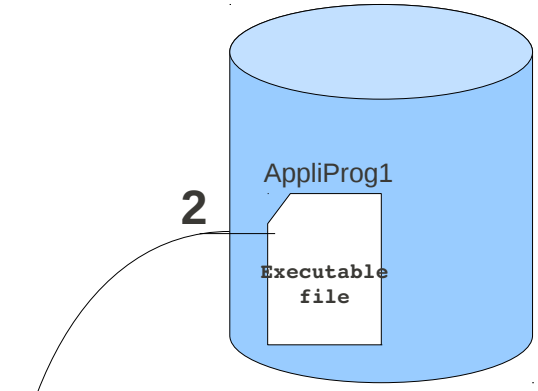


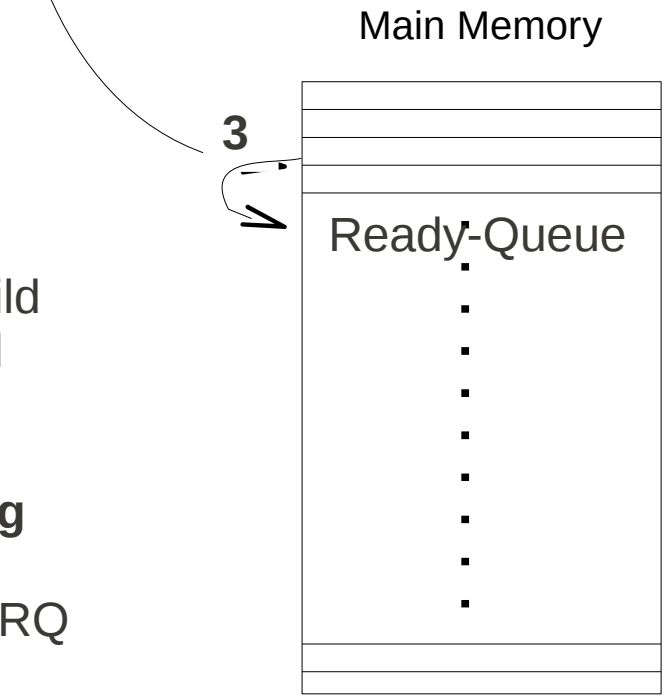


1. User launches “AppliProg1” either by CLI/GUI/...
2. The CLI/GUI process **fork/clones** to execute AppliProg1 as its child Now this process is in **NEW state.** Here prog loads into memory and process gets its PCB
3. LTS moves into **Ready state** into Ready-queue
4. STS selects this “AppliProg1” for execution marking it as “**Running State**” based on the scheduling policy (ex: FCFS, PBS, SJF, etc.)

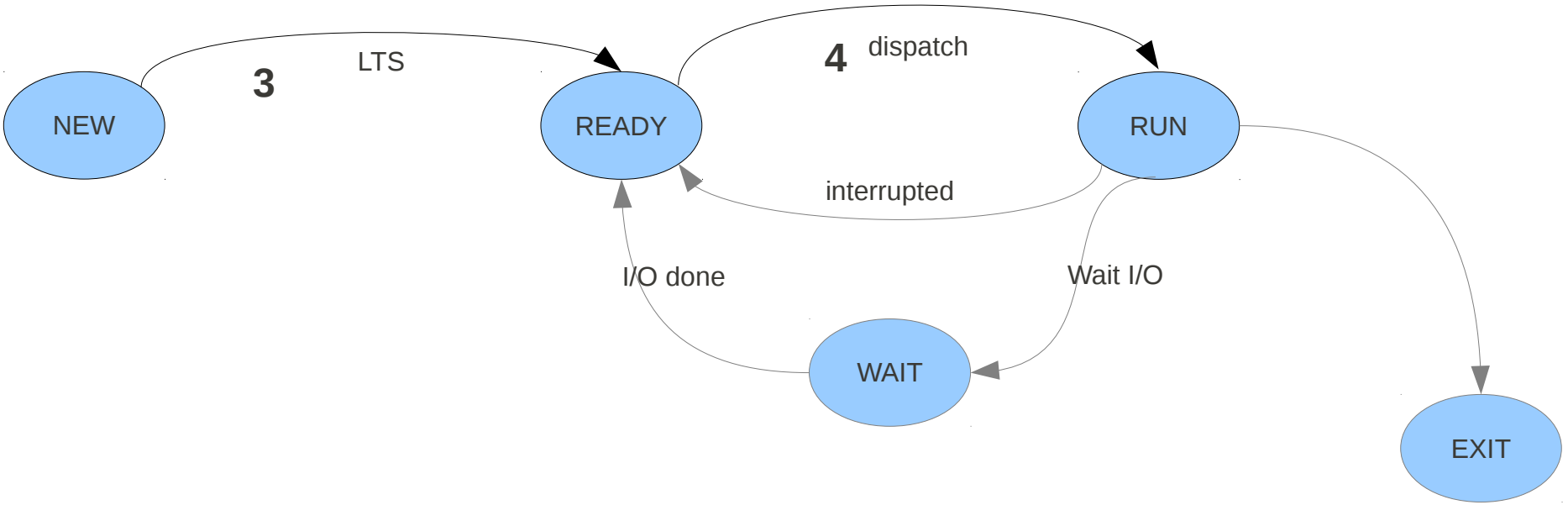








1. User launches “AppliProg1” either by CLI/GUI/...
2. The CLI/GUI process **fork/clones** to execute AppliProg1 as its child Now this process is in **NEW state.** Here prog loads into memory and process gets its PCB
3. LTS moves into **Ready state** into Ready-queue
4. STS selects this “AppliProg1” for execution marking it as “**Running State**” based on the scheduling policy (ex: FCFS, PBS, SJF, etc.)
5. The rest of process state diagram corresponds to I/O waitng and IRQ activities as discussed earlier in class.



1. User launches “AppliProg1” either by CLI/GUI/...
2. The CLI/GUI process **fork/clones** to execute AppliProg1 as its child Now this process is in **NEW state.** Here prog loads into memory and process gets its PCB
3. LTS moves into **Ready state** into Ready-queue
4. STS selects this “AppliProg1” for execution marking it as “**Running State**” based on the scheduling policy (ex: FCFS, PBS, SJF, etc.)
5. The rest of process state diagram corresponds to I/O waitng and IRQ activities as discussed earlier in class.