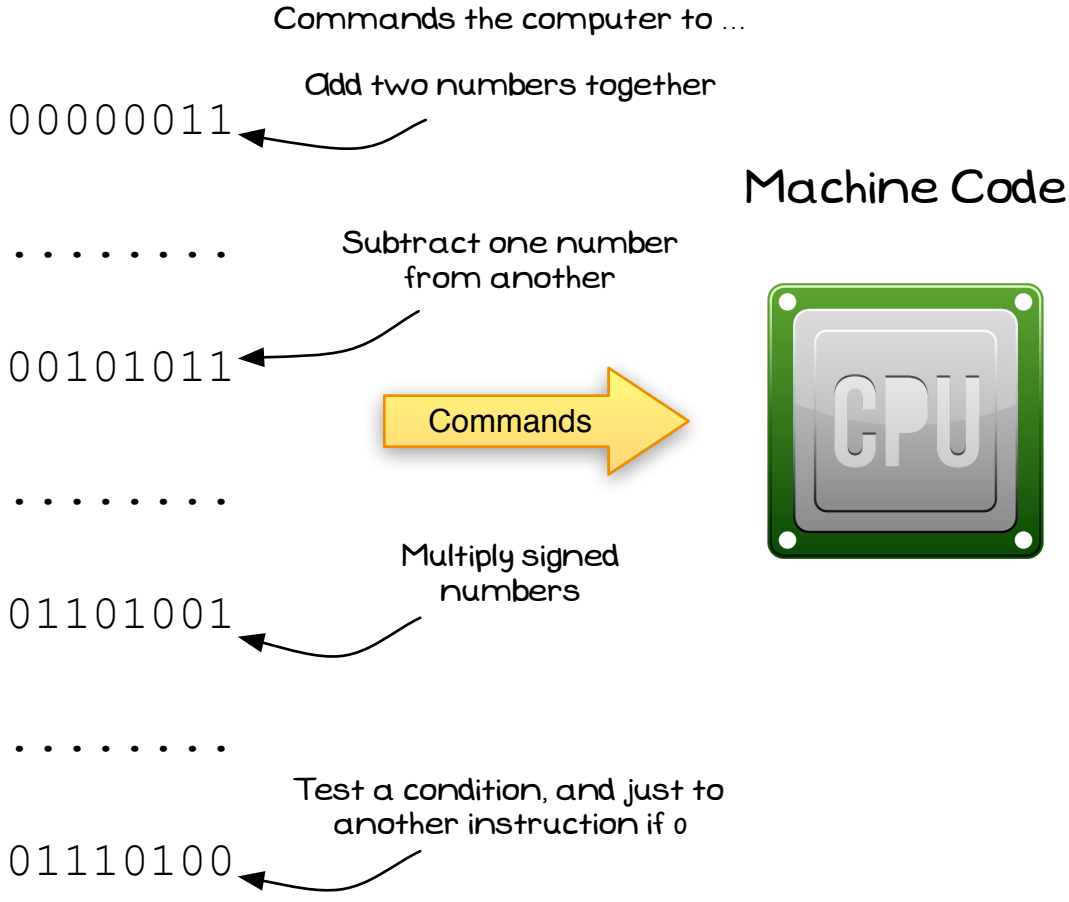
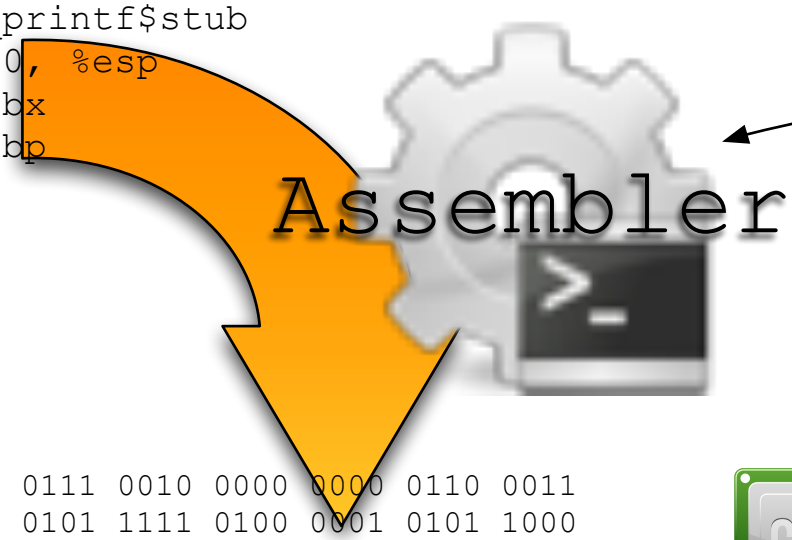

Commands the computer to ...
Add two numbers together
00000011
Machine Code
.....
Subtract one number from another
00101011
Commands
CPU
.....
Multiply signed numbers
01101001
.....
Test a condition, and just to another instruction if 0
01110100

Assembly Code

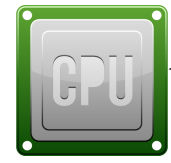
```
  .cstring
LC0:
  .ascii "Hello\0"
  .text
.globl _main
_main:
  pushl   %ebp
  movl %esp, %ebp
  pushl   %ebx
  subl $20, %esp
  call ___i686.get_pc_thunk.bx
"L000001$pb":
  leal LC0-"L000001$pb"(%ebx), %eax
  movl %eax, (%esp)
  call L_printf$stub
  addl $20, %esp
  popl %ebx
  popl %ebp
  ret
```

Symbolic names for the machine code instructions

Assembler converts Assembly Code to Machine Code

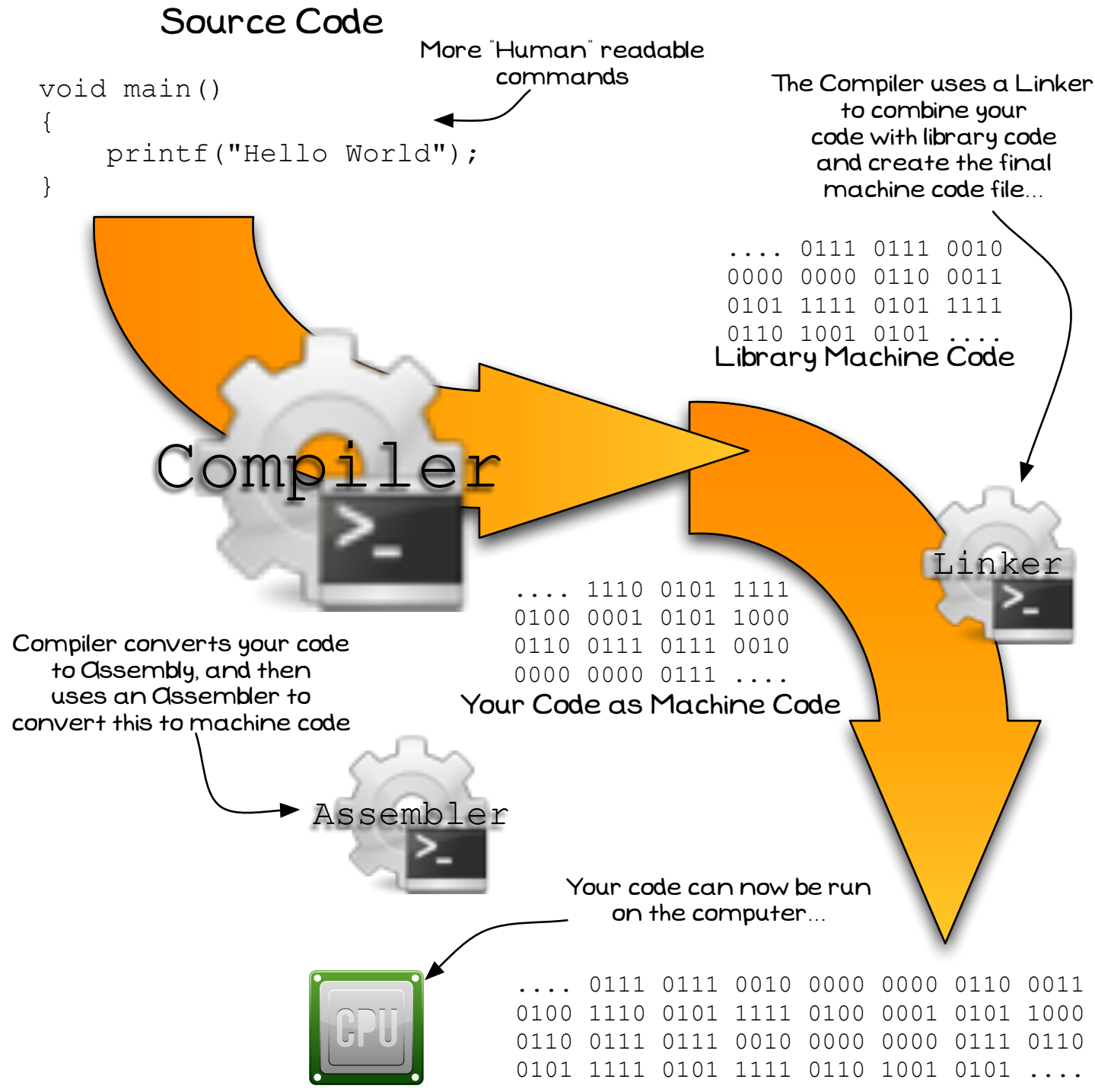


Only the machine code can be executed by the computer

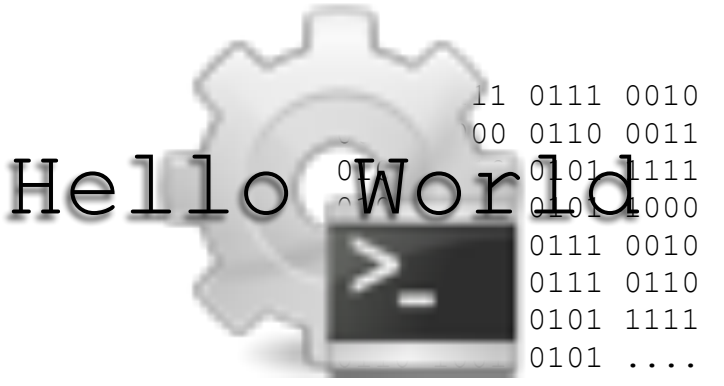



```
.... 0111 0111 0010 0000 0000 0110 0011
0100 1110 0101 1111 0100 0001 0101 1000
0110 0111 0111 0010 0000 0000 0111 0110
0101 1111 0101 1111 0110 1001 0101 ....
```

Machine Code

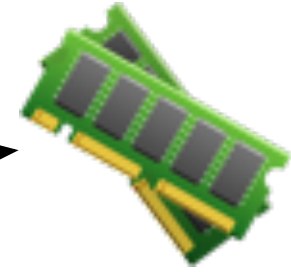

Source Code
More "Human" readable commands
void main()
{
printf("Hello World");
}
The Compiler uses a Linker to combine your code with library code and create the final machine code file...
.... 0111 0111 0010
0000 0000 0110 0011
0101 1111 0101 1111
0110 1001 0101 ....
Library Machine Code
Compiler
Linker
.... 1110 0101 1111
0100 0001 0101 1000
0110 0111 0111 0010
0000 0000 0111 ....
Your Code as Machine Code
Compiler converts your code to Assembly, and then uses an Assembler to convert this to machine code
Assembler
Your code can now be run on the computer...
CPU
.... 0111 0111 0010 0000 0000 0110 0011
0100 1110 0101 1111 0100 0001 0101 1000
0110 0111 0111 0010 0000 0000 0111 0110
0101 1111 0101 1111 0110 1001 0101 ....

A program is an executable file



It contains machine code instructions to tell the computer what to do when it is run

The program's file will exist on your computer's disk

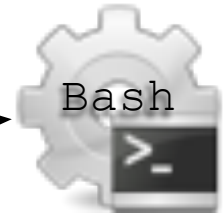


The file is loaded into memory before it can start executing

The Terminal is a program.

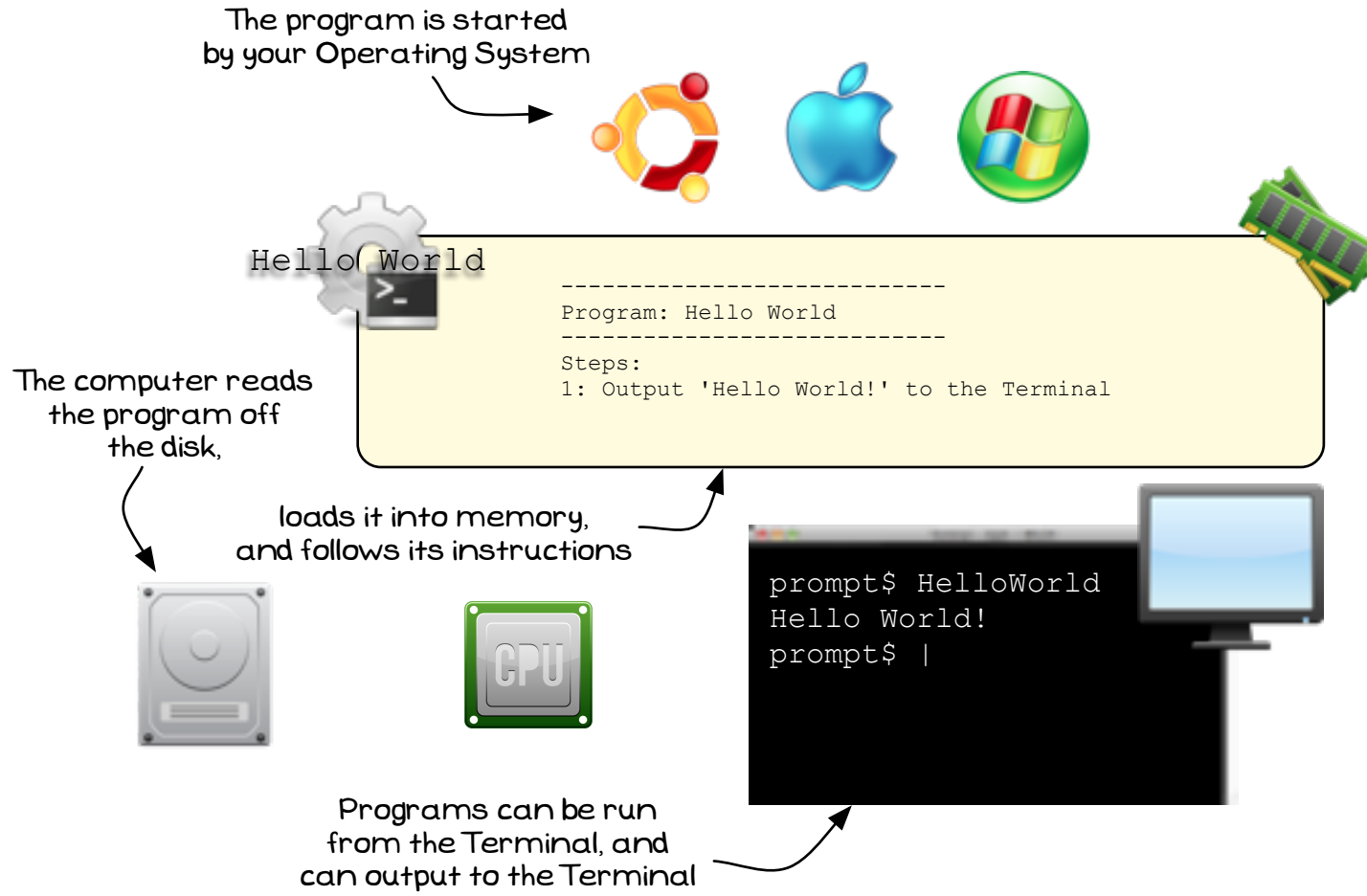
It creates an Window into
which programs can output tex,

and from which it can read input



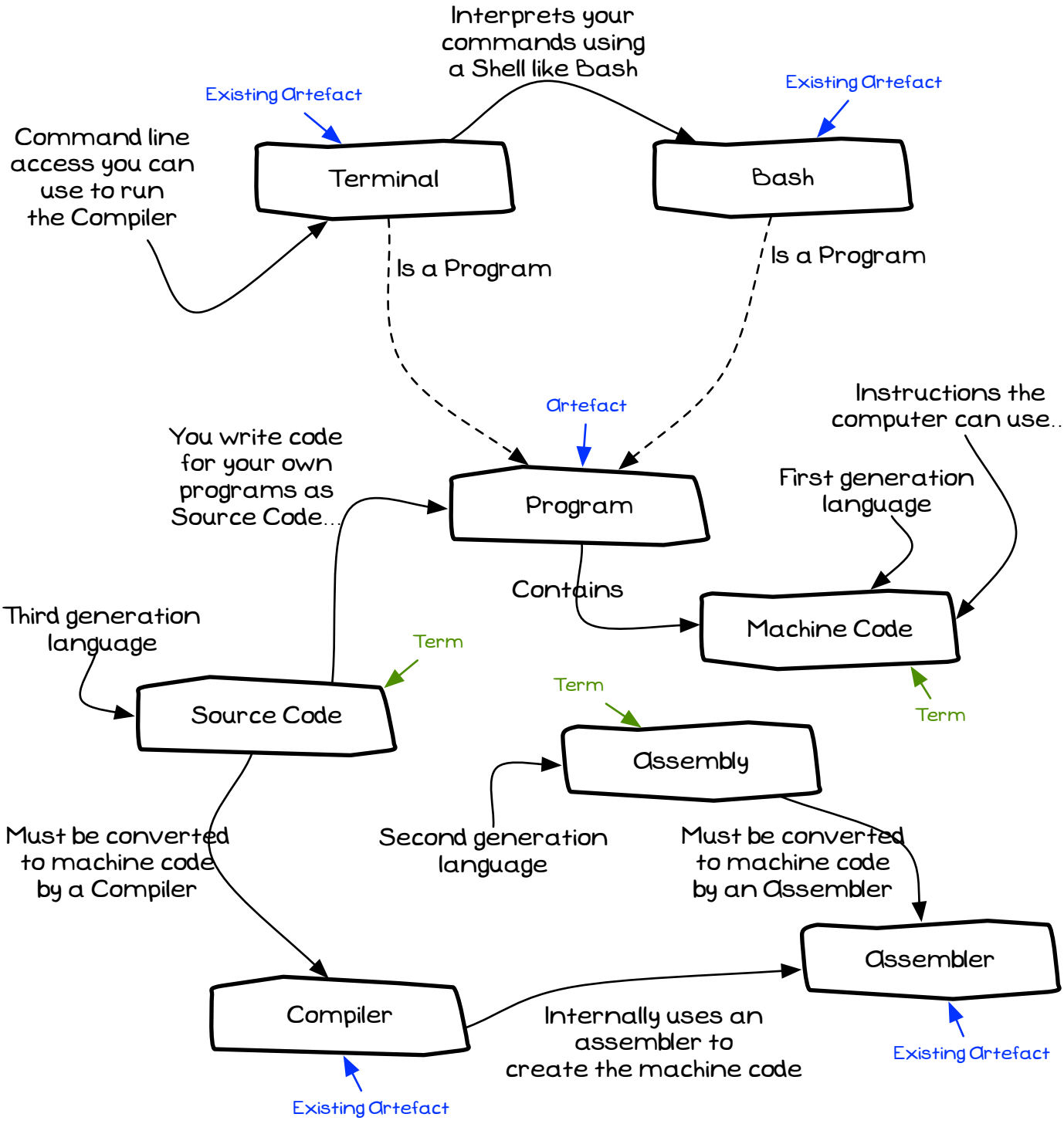
Bash is a shell - a program
that interprets your commands
and commands the computer to
perform actions...

Bash is run for you when the
Terminal starts, it will show you
a command prompt where you
can type your commands


The program is started by your Operating System
Hello World

Program: Hello World

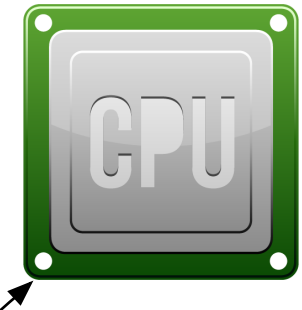
Steps:
1: Output 'Hello World!' to the Terminal
The computer reads the program off the disk,
loads it into memory, and follows its instructions
CPU
prompt$ HelloWorld
Hello World!
prompt$ |
Programs can be run from the Terminal, and can output to the Terminal


Interprets your commands using a Shell like Bash
Existing Artefact
Existing Artefact
Command line access you can use to run the Compiler
Terminal
Bash
Is a Program
Is a Program
Artefact
Instructions the computer can use...
You write code for your own programs as Source Code...
Program
First generation language
Contains
Machine Code
Third generation language
Term
Term
Source Code
Term
Assembly
Must be converted to machine code by a Compiler
Second generation language
Must be converted to machine code by an Assembler
Assembler
Compiler
Internally uses an assembler to create the machine code
Existing Artefact
Existing Artefact

Intelligent

Unintelligent

The Developer codes the "Intelligence" into the Program



The program runs on the computer, and makes it look good...