

Engine - Currently, V8 is the most popular and common. Written on C++.

3. These tokens formed something called Abstract Syntax Tree / AST/

https://astexplorer.net

6.Compiler - doesn't translate on the fly. It works ahead of time to create translation of code which we have just written. Compiles down to language that can be understood by our machines.

So compiler tries to understand what we want to do and compiles our language to lower level language such as machine code.

The compile is not perfect and it can make mistakes. If it does something unexpected then it does something called D optimization which takes even longer time to revert back to the interpreter.

Pros:

+ if compiler sees repeating code, he simplified it, and eventually run faster.

Cons:

- little more time to start because it has to go through the compilation step at the beginning

4.Interpreter/called Ignition on V8/ - translate and read files line by line on the fly.

Pros:

- + quick to get up and running Cons:
- running the same code more than once even though it gives the same result it can get really slow
- Doesn't do any optimizations.

5.Profiler - Also called a monitor Watches our code when it is running and make notes how we can optimize our code.

For example, how many times it is being run. What types are used etc...

4.1 Bytecode

- Not as low level as machine code, but it is code that is able to be interpreted by JS.

JIT Compiler
/Just In Time /
Combine between Interoreter and
compiler to make engine faster

Optimize the code