

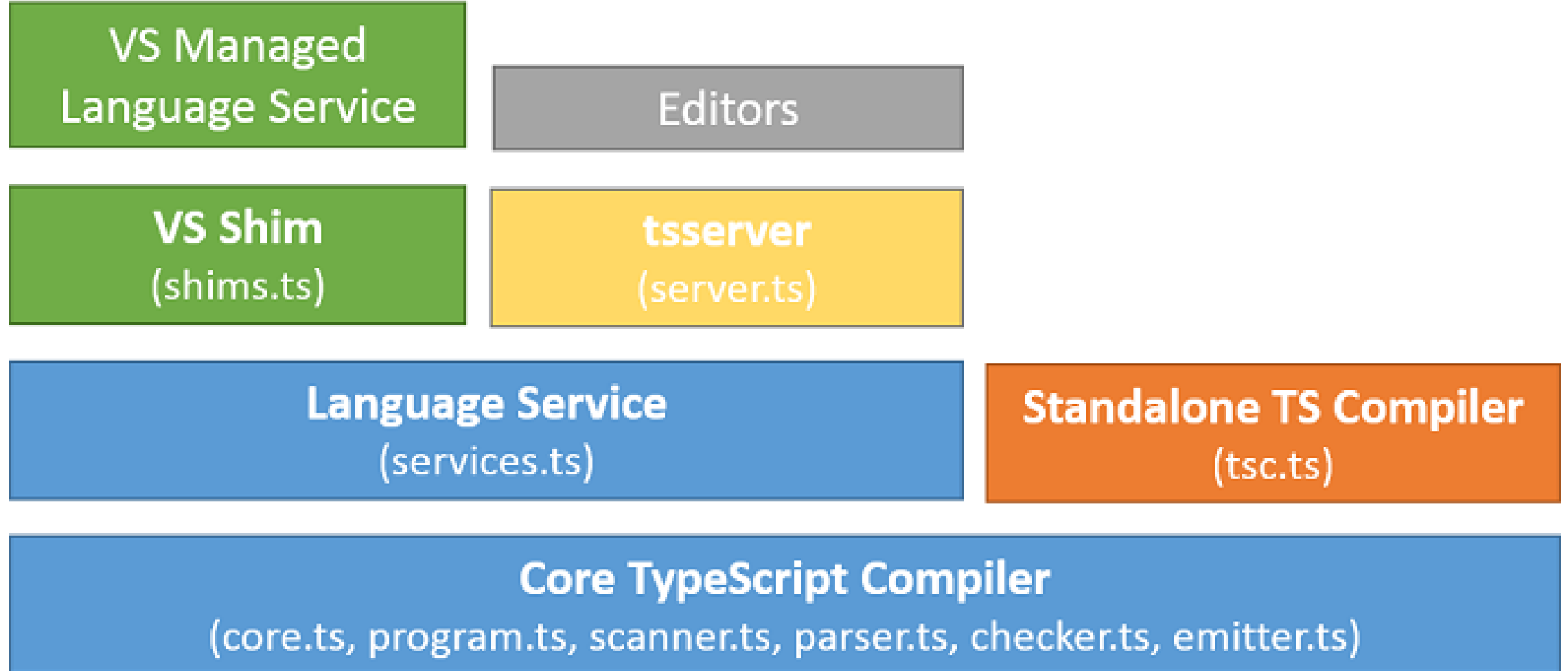


TypeScript compiler API 101

COSCon'20 Workshop

Please make sure you have followed the instruction in <https://github.com/Jack-Works/oct-2020-typescript-compiler-101> before the workshop starts.

Architectural overview



How a project is represented in the compiler?

Program

SourceFile (index.ts)

SourceFile (utils.ts)

SourceFile (cli.ts)

SourceFile (io.ts)

Then what about a single file?

```
1 import fs from 'fs/promise'
2 fs.readFile('./package.json')
```

- ▼ **SourceFile**
 - ▼ ImportDeclaration
 - ▼ ImportClause
 - Identifier
 - StringLiteral
 - ▼ ExpressionStatement
 - ▼ CallExpression
 - ▼ PropertyAccessExpression
 - Identifier
 - Identifier
 - StringLiteral
 - EndOfFileToken

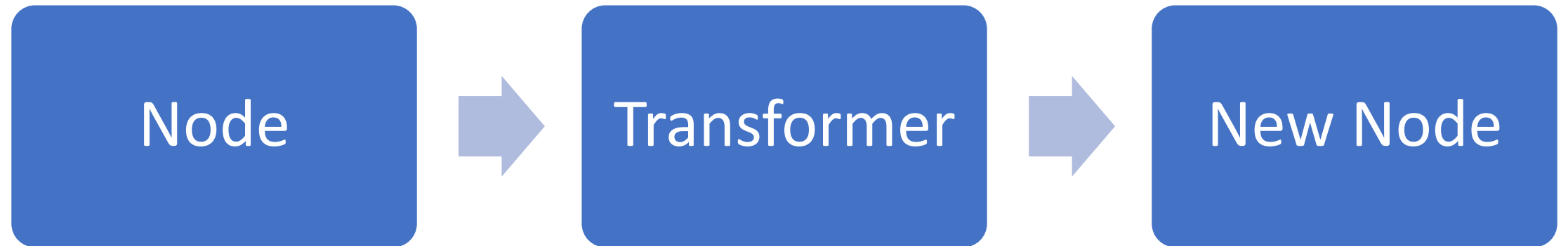
What we're going to craft today

AST transformer

- How you generally use it?
- - By [configuring ts-loader](#) in webpack

```
- - 1  import ts from 'typescript'
    2  ✓ ts.transpileModule('export default 0', {
    3  ✓    transformers: {
    4      // run before built-in TS ⇒ JS transformer
    5      before: [],
    6      // run after built-in TS ⇒ JS transformer
    7      after: [],
    8      // run after declaration files generated
    9      afterDeclarations: [],
   10    },
   11  })
```

But what is a transformer?



Emitting in TypeScript ([source code](#))

